AFWAL-TR-86-3094

LAMINAR BOUNDARY LAYER ALONG INVISCID SURFACE STREAMLINES OVER INCLINED AXISYMMETRIC BODIES IN INCOMPRESSIBLE FLOW

FRED R. DEJARNETTE

DEPT. OF MECHANICAL AND AEROSPACE ENGINEERING
NORTH CAROLINA STATE UNIVERSITY
RAYLEIGH, NC 27695-7910
(919) 737-2365

FEBRUARY, 1988

FINAL REPORT FOR PERIOD AUG, 1981 THROUGH MAY, 1983

FLIGHT DYNAMICS LABORATORY
AIR FORCE AERONAUTICAL LABORATORIES
AIR FORCE SYSTEM COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO 45433-6553

NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely Government-related procurement, the United States Government incurs no responsibility or any obligation whatsoever. The fact that the government may have formulated or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication, or otherwise in any manner construed, as licensing the holder, or any other person or corporation; or as conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

This report is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.

This technical report has been reviewed and is approved for publication.

William H. Lane

W. H. Lane, Project Engineer
Flight Control Division

David K. Bowser

David K. Bowser, Chief
Control Dynamics Branch
Flight Control Division

FOR THE COMMANDER

Eugene A. Smith

Eugene A. Smith, Lt Col, USAF
Chief, Flight Control Division

If your address has changed, if you wish to be removed from our mailing list, or if the addressee is no longer employed by your organization please notify WRDC/FIGC, WPAFB, OH 45433-6553 to help us maintain a current mailing list.

Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

1a. REPORT SECURITY CLASSIFICATION: Unclassified	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT: Approved for Public Release; Distribution Unlimited
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S)	5. MONITORING ORGANIZATION REPORT NUMBER(S): AFWAL-TR-86-3094
6a. NAME OF PERFORMING ORGANIZATION: Mechanical & Aerospace Eng Dept N Carolina State Univ at Raleigh; 6b. OFFICE SYMBOL (If applicable)	7a. NAME OF MONITORING ORGANIZATION: Flight Dynamics Laboratory (AFWAL/FIGC) Air Force Wright Aeronautical Laboratories
6c. ADDRESS (City, State, and ZIP Code): Raleigh NC 27650	7b. ADDRESS (City, State, and ZIP Code): Wright-Patterson AFB OH 45433-6553
8a. NAME OF FUNDING/SPONSORING ORGANIZATION: Air Force Office of Scientific Research; 8b. OFFICE SYMBOL (If applicable): AFOSR/NA	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER: F33615-81-K-3625
8c. ADDRESS (City, State, and ZIP Code): Bolling AFB DC 20332	10. SOURCE OF FUNDING NUMBERS: PROGRAM ELEMENT NO. 61102F; PROJECT NO 2307; TASK NO N3; WORK UNIT ACCESSION NO. 24

11. TITLE (Include Security Classification)
Laminar Boundary Layer Along Inviscid Surface Streamlines Over Inclined Axisymmetric Bodies in Incompressible Flow

12. PERSONAL AUTHOR(S)
Fred R. DeJarnette and Lee H. Kania

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT
Final	FROM Aug 81 TO May 83	1988 February	145

16. SUPPLEMENTARY NOTATION

17. COSATI CODES		
FIELD	GROUP	SUB-GROUP
20	04	

18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
Theoretical and experimental fluid dynamics
Computational fluid dynamics
Three-dimensional flow

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

A computer program has been developed to calculate laminar boundary layers along inviscid surface streamlines over inclined axisymmetric bodies. The inviscid surface streamlines are computed from experimental pressure data using a doubly quadratic spline interpolation technique if an analytical potential solution is not provided. By application of the axisymmetric analog, or small cross flow assumption, the properties of this three-dimensional flow field may be approximated by solving the axisymmetric boundary layer equations over an equivalent axisymmetric body for each inviscid surface streamline. Hall's and Blottner's boundary layer solution methods, both of which utilize a finite difference technique to obtain the local velocity profile, have been incorporated into the computational code.

Results are presented for a sphere, an ellipsoid of revolution with thickness ratio of 1/4 at 12 and 30 degrees angle of attack, and a sphere-ogive-cylinder configuration at 45 degrees angle of attack. The latter case utilized experimental pressure (con't on back)

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT	21. ABSTRACT SECURITY CLASSIFICATION
☐ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☒ DTIC USERS	Unclassified

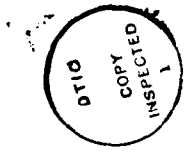
22a. NAME OF RESPONSIBLE INDIVIDUAL	22b. TELEPHONE (Include Area Code)	22c. OFFICE SYMBOL
William H. Lane	(513) 255-8486	AFWAL/FIGC

DD Form 1473, JUN 86 — Previous editions are obsolete — SECURITY CLASSIFICATION OF THIS PAGE

UNCLASSIFIED

19. ABSTRACT (con't)

data while an analytical potential solution was available for the other configurations. A technique for calculating the streamline metric along an inviscid surface streamline was developed. Boundary layer profiles were calculated along the streamlines from the stagnation point to the separation point. Calculations using the axisymmetric analog and three-dimensions boundary layer equations were compared for the particular case of an ellipsoid of revolution at angle of attack.



FOREWORD

This report describes the method and computer program for calculating laminar boundary-layer properties over bodies of revolution at large incidence in subsonic flow. With minor changes, it was also Mr. Lee A. Kania's thesis for the Master of Science Degree in Mechanical Engineering at North Carolina State University in 1983.

This work was performed by Dr. Fred R. DeJarnette, principal investigator, and Mr. Lee A. Kania, graduate research assistant, Mechanical and Aerospace Engineering Department, North Carolina State University, Raleigh, North Carolina 27695-7910. It was supported under Procurement Instrument Identification Number (Contract Number) F33615-81-K-3625 with the Air Force Wright Aeronautical Laboratories, Wright-Patterson Air Force Base, Ohio 45433-6553. The subject contract was initiated under Air Force Flight Dynamics Laboratory Project 2307, Task 2307N324, on 17 August 1981 and was effectively concluded in May 1983. Mr. William H. Lane, AFWAL/FIGC, was the Air Force Project Engineer for the study. Comments may be directed to him at (513)255-8486, or in writing at the above address.

Copies of this report can be obtained from the National Technical Information Service (NTIS).

This report was submitted in 1983 and revised in 1987.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SECTION 1

INTRODUCTION

There are at present numerous methods available for calculating boundary layers over two-dimensional and axisymmetric bodies at zero angle of attack (Refs. 1 and 2). Methods for calculating three-dimensional boundary layers (Refs. 3, 4 and 5) are not as numerous due to the fact that they have only recently come under investigation. These techniques are, at present, limited in application. Generally, fully three-dimensional methods require considerable storage and computational time on existing digital computers. To compound the problem, the potential solution in most cases cannot be described through a simple analytical expression.

A relatively simple, approximate method for calculating three-dimensional boundary layer properties is the axisymmetric analog. In this method the boundary-layer equations are written in a streamline coordinate system and the cross flow velocity is assumed to be zero. This reduces the three-dimensional boundary layer equations to a form that is identical to those for axisymmetric flow, provided that (1) the distance along an inviscid surface streamline is interpreted as distance along an "equivalent axisymmetric body," and (2) the metric coefficient that describes the spreading of the streamlines is interpreted as the radius of the equivalent body. This allows any existing axisymmetric boundary-layer program to be used to compute the approximate three-dimensional boundary-layer properties. By considering multiple streamline paths, an entire surface can be covered.

The major difficulties in applying the axisymmetric analogue are the calculation of the inviscid surface streamlines and the corresponding scale factor. References (7) and (8) provide two methods with which to trace inviscid surface streamlines from surface pressures. These two approaches are basically identical in that each requires the integration of a first-order differential equation to yield the streamline angle (the angle between a streamline and body meridian).

In Reference (7) Vollmers proposes that a shooting technique be employed to determine an initial value for the streamline angle. He contends that a valid initial value may be obtained near the stagnation point if the streamline angle approaches the correct limit during the upstream integration from a given point on the body. For a nonspherical nose, the streamline geometry is such that the streamline angle is either 0 or 180 degrees in the limit at the stagnation point (Ref. 9). Vollmers, however, fails to account for the behavior of the scale factor during the upstream integration. There is a possibility that the scale factor may tend to zero during the integration despite the fact that the streamline angle may approach the proper limiting value. A scale factor of zero implies that the streamlines cross at a particular point and this is a physical impossibility. This condition can occur when approximate surface pressures are used. If the inviscid surface velocity components were known, the streamlines could be calculated more easily and there would be no possibility of streamline crossing.

In Reference (8), DeJarnette attacks this problem by describing the streamline geometry analytically in the stagnation region. Outside this region, the inviscid surface streamlines are calculated from the surface

pressure distribution. This is the approach employed in the present method for instances in which an analytical potential solution is not available.

The method used to calculate the "equivalent radius" along a streamline follows from a method developed by DeJarnette in Reference (9) and is included in this study. DeJarnette has previously used a two streamline approach to determine the scale factor, but this necessitates the calculation of a second or auxiliary streamline. In the present method, the scale factor is calculated along a single streamline from the solution of two first-order, auxiliary differential equations which are functions of the surface pressure distribution and body geometry.

The present method employs Hall's (Ref. 4) and Blottner's (Ref. 2) methods to obtain a solution to the axisymmetric boundary-layer equations. Hall applies a Crank-Nicholson differencing technique to the nondimensionalized equations. The body radius, which appears only in the continuity equation, is replaced by the scale factor in accordance with the axisymmetric analogue. Blottner employs the same differencing technique to the axisymmetric boundary-layer equations written in transformed variables. In this case the body radius appears only in the definition of the transformed variables and is likewise replaced with the scale factor. The body radius does not appear explicitly in the transformed boundary-layer equations. The development of each of these methods is included in this study. In the development of the boundary-layer code, various velocity profile convergence tests and boundary-layer edge tests are investigated also.

Any axisymmetric configuration may be input to the program as long as an analytical expression for the pressure distribution is provided. In the event that only experimental pressure data are available, the input geometry is restricted to spherically capped bodies due to the limitations of the techniques used to represent the surface pressure distribution. On the spherical cap the pressure distribution is represented by a Fourier cosine series while on the remainder of the body, a doubly quadratic spline is used to model the experimental pressure data. The body geometry may be expressed in either English or SI units, or in nondimensional form.

Results from the computer program are presented for a sphere, ellipsoid of revolution with thickness ratio 1/4 and a sphere-ogive-cylinder configuration as example applications of the computational method.

SECTION 2

INVISCID SURFACE STREAMLINES

The axisymmetric analogue concept employed in the computer program effectively reduces the three-dimensional nature of the boundary layer to that of an axisymmetric one along an inviscid surface streamline. This necessitates the calculation of the equivalent radius of the newly defined axisymmetric body. The equivalent radius or scale factor is the metric for the coordinate β normal to the streamline on the body surface and is calculated along inviscid surface streamlines. The scale factor is an indicator of the physical spacing between streamlines. A scale factor that is increasing indicates that the streamlines are diverging and thus the equivalent radius is increasing.

The method of DeJarnette (Ref. 8) is used to trace the inviscid surface streamlines. In this method, the body geometry is expressed in terms of the unit vectors, $\hat{e}_x$, $\hat{e}_r$ and $\hat{e}_\phi$ which form an orthogonal cylindrical coordinate system. Unit vector $\hat{e}_x$ is parallel to the body axis, unit vector $\hat{e}_r$ is in the radial direction and normal to the body axis. The third unit vector in this system, $\hat{e}_\phi$, is in the circumferential direction (see Figure 2.1).

A second coordinate system which is oriented to the body surface is used to describe the surface streamlines. This system consists of the unit vectors, $\hat{e}_{11}$, $\hat{e}_n$, and $\hat{e}_\phi$. Unit vector $\hat{e}_n$ is normal to the body surface and is given by

$$\hat{e}_n = - \sin \Gamma \, \hat{e}_x + \cos \Gamma \, \hat{e}_r \quad . \tag{2.1}$$

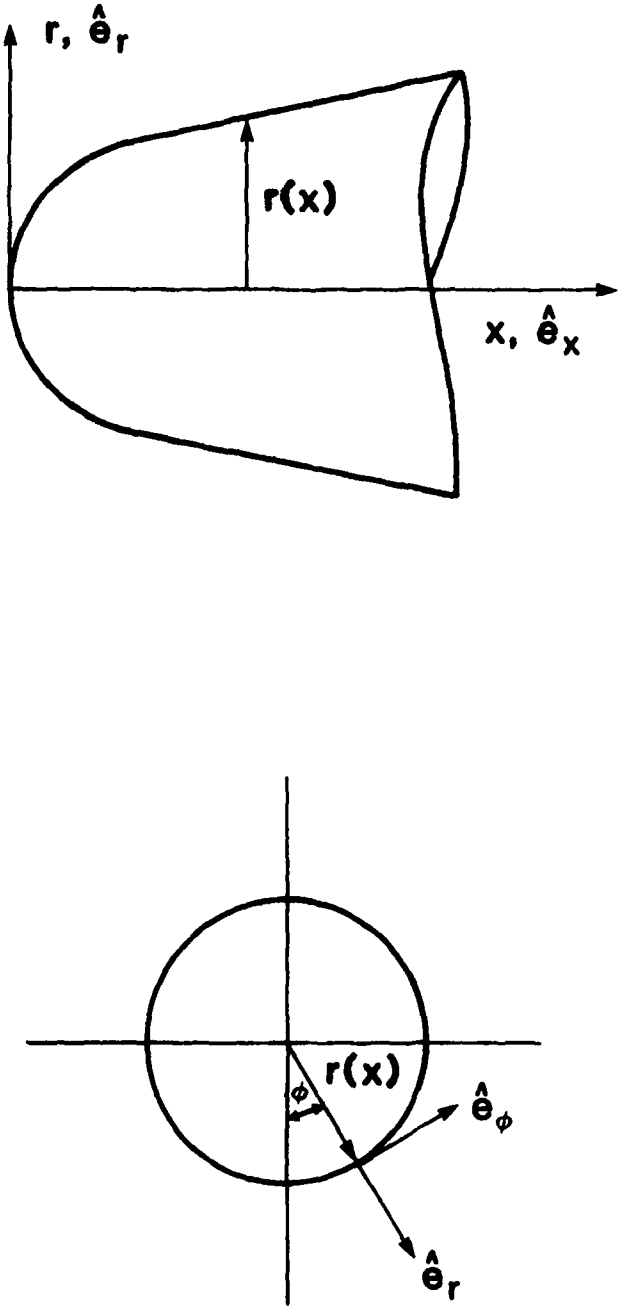


Figure 2.1. Body Geometry Coordinate System

Unit vector, $\hat{e}_{11}$, is tangent to the body surface and lies in a meridianal plane. This vector may be expressed as

$$\hat{e}_{11} = \cos \Gamma \, \hat{e}_x + \sin \Gamma \, \hat{e}_r \quad . \tag{2.2}$$

The angle Γ is the body angle and is a function of x only for an axisymmetric body (see Figure 2.2).

The streamline geometry is expressed in terms of unit vectors, $\hat{e}_s$, $\hat{e}_\beta$ and $\hat{e}_n$ which also form an orthogonal coordinate system. Unit vector $\hat{e}_s$ is along a streamline and tangent to the body surface. Unit vector $\hat{e}_\beta$ is normal to the streamline and also tangent to the body surface. Unit vector $\hat{e}_n$ is used in common with the previous coordinate system. Since the streamline is projected on the body surface, the component along $\hat{e}_n$ is zero and a streamline will thus lie in the tangent plane generated by unit vectors $\hat{e}_s$ and $\hat{e}_\beta$. This is the identical plane generated by $\hat{e}_{11}$ and $\hat{e}_\phi$. The streamline angle θ is defined to be the angle between unit vectors $\hat{e}_s$ and $\hat{e}_{11}$ (see Figure 2.3). This angle is the inclination of the streamline relative to a body meridian. The streamline unit vectors may then be written in terms of the body geometry unit vectors as

$$\hat{e}_s = (\cos \theta \cos \Gamma) \, \hat{e}_x + (\cos \theta \sin \Gamma) \, \hat{e}_r + (\sin \theta) \, \hat{e}_\phi \tag{2.3}$$

and

$$\hat{e}_\beta = -(\sin \theta \cos \Gamma) \, \hat{e}_x - (\sin \theta \sin \Gamma) \, \hat{e}_r + (\cos \theta) \, \hat{e}_\phi \quad . \tag{2.4}$$

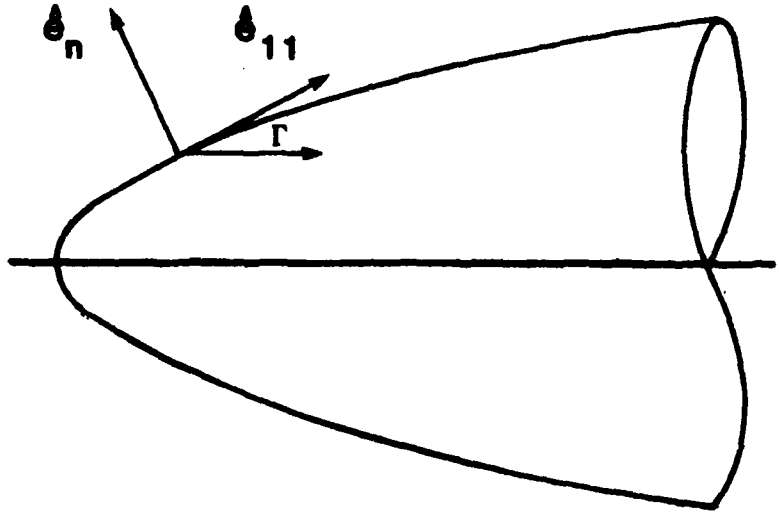


Figure 2.2. Surface Coordinate System

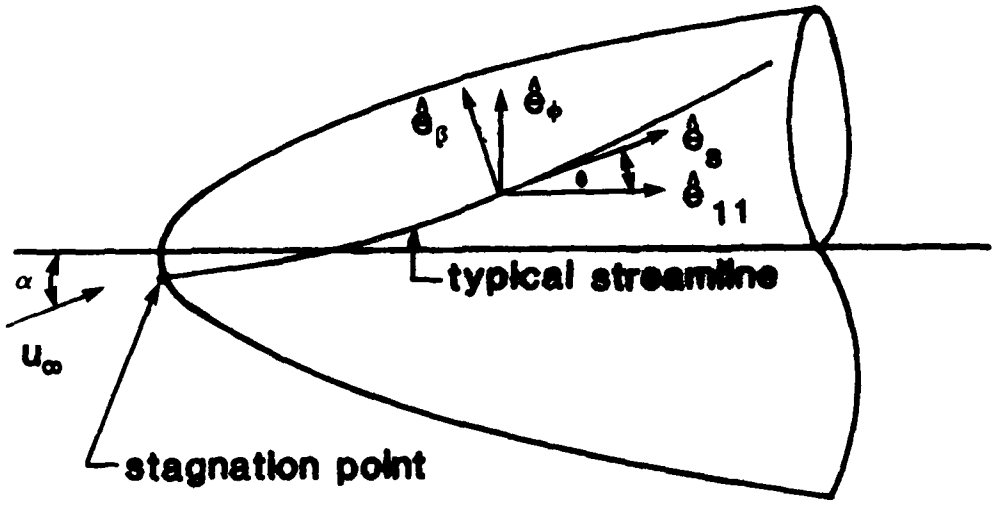


Figure 2.3. Streamline Coordinate System

DeJarnette (Ref. 9) then constructs the transformation operators which relate the streamline partial derivatives to the cylindrical coordinate derivatives. The operators may be written as

$$\frac{1}{h}\frac{\partial}{\partial\beta} = (\hat{e}_x \cdot \hat{e}_\beta)\frac{\partial}{\partial x} + \frac{(\hat{e}_\phi \cdot \hat{e}_\beta)}{r}\frac{\partial}{\partial\phi} \tag{2.5}$$

and

$$\frac{D}{DS} = (\hat{e}_x \cdot \hat{e}_s)\frac{\partial}{\partial x} + \frac{(\hat{e}_\phi \cdot \hat{e}_s)}{r}\frac{\partial}{\partial\phi} \tag{2.6}$$

where D/DS is a derivative along a streamline and $\frac{1}{h}\frac{\partial}{\partial\beta}$ is a derivative normal to a streamline and on the body surface. Substituting the expressions for the unit vectors yields

$$\frac{1}{h}\frac{\partial}{\partial\beta} = -\sin\theta\cos\Gamma\frac{\partial}{\partial x} + \frac{\cos\theta}{r}\frac{\partial}{\partial\phi} \tag{2.7}$$

and

$$\frac{D}{DS} = \cos\theta\cos\Gamma\frac{\partial}{\partial x} + \frac{\sin\theta}{r}\frac{\partial}{\partial\phi} \quad . \tag{2.8}$$

Application of Equation (2.8) yields

$$\frac{Dx}{DS} = \cos\theta\cos\Gamma \tag{2.9}$$

and

$$\frac{D\phi}{DS} = \frac{\sin\theta}{r} \quad . \tag{2.10}$$

These differential equations may be numerically integrated to give the axial and circumferential position along a streamline when θ is known (the initial values for all differential equations will be discussed later in this section). If the potential solution were known in analytic form, the angle θ could be determined from it. When only pressure data are available, θ must be calculated from a differential equation.

DeJarnette derives the differential equation for the streamline angle θ from the application of Euler's equation on the surface of the body. In vector form Euler's equation is

$$\frac{D\vec{U}}{Dt} = -\frac{\nabla p}{\rho} \quad . \tag{2.11}$$

The convective term may be recast in streamline coordinates as

$$U \frac{DU}{DS} \hat{e}_s + U^2 \frac{D\hat{e}_s}{DS} \quad . \tag{2.12}$$

The pressure gradient in streamline coordinates may also be written as

$$\nabla p = \frac{Dp}{DS} \hat{e}_s + \frac{1}{h} \frac{\partial p}{\partial \beta} \hat{e}_\beta + \frac{\partial p}{\partial n} \hat{e}_n \quad . \tag{2.13}$$

Euler's equation may then be written as

$$U \frac{DU}{DS} \hat{e}_s + U^2 \frac{D\hat{e}_s}{DS} = -\frac{1}{\rho} \left\{ \frac{Dp}{DS} \hat{e}_s + \frac{1}{h} \frac{\partial p}{\partial \beta} \hat{e}_\beta + \frac{\partial p}{\partial n} \hat{e}_n \right\} . \tag{2.14}$$

The scalar product of $\hat{e}_\beta$ with this equation gives

$$U^2 \frac{D\hat{e}_s}{DS} \cdot \hat{e}_\beta = -\frac{1}{\rho h} \frac{\partial p}{\partial \beta} \quad . \tag{2.15}$$

The scalar product of $\hat{e}_\beta$ with the derivative of the expression for $\hat{e}_s$ yields (Ref. 9)

$$d\hat{e}_s \cdot \hat{e}_\beta = d\theta + \sin \Gamma \, d\phi \quad . \tag{2.16}$$

Substitution of this equation into (2.15) yields

$$\frac{D\theta}{DS} + \sin \Gamma \frac{D\phi}{DS} = - \frac{1}{\rho U^2} \frac{1}{h} \frac{\partial p}{\partial \beta} \quad . \tag{2.17}$$

Application of the previously defined transformation operators can be used to write this equation as

$$\frac{D\theta}{DS} = - \frac{1}{2} \left(\frac{u_\infty}{u_e} \right)^2 \left\{ - \sin \theta \cos \Gamma \frac{\partial C_p}{\partial x} + \frac{\cos \theta}{r} \frac{\partial C_p}{\partial \phi} \right\} - \frac{\sin \Gamma \sin \theta}{r} \quad . \tag{2.18}$$

Equations (2.9), (2.10) and (2.18) can be numerically integrated to fully describe the geometry of the streamlines resulting from a given pressure distribution. If the potential solution is expressed analytically, θ may also be expressed analytically and only Equations (2.9) and (2.10) need be integrated.

The scale factor must be evaluated simultaneously during the streamline integration for use in the boundary-layer calculations. The technique employed here has not previously been used elsewhere but follows from a technique developed by DeJarnette (Ref. 9). If the streamline coordinate β is substituted into the transformation operators (2.7) and (2.8), the result is easily shown to be

$$\frac{1}{h} = - \sin \theta \cos \Gamma \left. \frac{\partial \beta}{\partial x} \right|_\phi + \frac{\cos \theta}{r} \left. \frac{\partial \beta}{\partial \phi} \right|_x \tag{2.19}$$

and

$$0 = \cos\theta \cos\Gamma \left.\frac{\partial\beta}{\partial x}\right|_{\phi} + \frac{\sin\theta}{r} \left.\frac{\partial\beta}{\partial\phi}\right|_{x} \quad , \tag{2.20}$$

respectively. Note that $\frac{D\beta}{DS} = 0$ since the coordinate β is constant along a streamline. Equation (2.20) may be solved for $\left.\frac{\partial\beta}{\partial x}\right|_{\phi}$ and then substituted into Equation (2.19) to yield

$$\frac{1}{h} = -\sin\theta \cos\Gamma \left(\frac{-\sin\theta}{r\cos\theta\cos\Gamma} \left.\frac{\partial\beta}{\partial\phi}\right|_{x}\right) + \frac{\cos\theta}{r} \left.\frac{\partial\beta}{\partial\phi}\right|_{x} \tag{2.21}$$

which reduces to

$$\frac{1}{h} = \frac{1}{r\cos\theta} \left.\frac{\partial\beta}{\partial\phi}\right|_{x} \tag{2.22}$$

or

$$h = r\cos\theta \left.\frac{\partial\phi}{\partial\beta}\right|_{x} \quad . \tag{2.23}$$

Note that since $\beta = \beta(x,\phi)$,

$$\left.\frac{\partial\beta}{\partial\phi}\right|_{x} = \frac{1}{\left.\frac{\partial\phi}{\partial\beta}\right|_{x}} \quad .$$

This expression necessitates the additional calculation of $\left.\frac{\partial\phi}{\partial\beta}\right|_{x}$ along a streamline in order to calculate the scale factor h.

The differential equation for $\left.\frac{\partial\phi}{\partial\beta}\right|_{x}$ may be obtained as follows. Equations (2.9) and (2.10) may be combined to give

$$\frac{D\phi}{Dx} = \frac{\tan\theta}{r\cos\Gamma} \quad . \tag{2.24}$$

Taking $\left.\frac{\partial}{\partial\phi}\right|_x$ of both sides of this equation yields

$$\frac{\partial}{\partial\phi}\left(\frac{D\phi}{Dx}\right) = \frac{\sec^2\theta}{r\cos\Gamma}\frac{\partial\theta}{\partial\phi} \tag{2.25}$$

since both r and Γ are functions of x only for an axisymmetric body. With $\beta = \beta(x,\phi)$,

$$\left.\frac{\partial}{\partial\phi}\right|_x = \left.\frac{\partial\beta}{\partial\phi}\right|_x \left.\frac{\partial}{\partial\beta}\right|_x \tag{2.26}$$

and

$$\frac{D\phi}{Dx} = \left.\frac{\partial\phi}{\partial x}\right|_\beta \tag{2.27}$$

the derivative of Equation (2.24) with respect to ϕ may be written as

$$\frac{\partial}{\partial\phi}\left(\frac{D\phi}{Dx}\right) = \left.\frac{\partial\beta}{\partial\phi}\right|_x \left.\frac{\partial}{\partial\beta}\right|_x \left.\left(\frac{\partial\phi}{\partial x}\right)\right|_\beta = \frac{\sec^2\theta}{r\cos\Gamma}\left.\frac{\partial\theta}{\partial\phi}\right|_x . \tag{2.28}$$

This in turn may be rewritten as

$$\frac{\partial}{\partial\phi}\left(\frac{D\phi}{Dx}\right) = \frac{1}{\left.\frac{\partial\phi}{\partial\beta}\right|_x}\frac{D}{Dx}\left.\left(\frac{\partial\phi}{\partial\beta}\right)\right|_x \tag{2.29}$$

or

$$\frac{D}{Dx}\left(\ell n\left.\frac{\partial\phi}{\partial\beta}\right|_x\right) = \frac{1}{r\cos^2\theta\cos\Gamma}\left.\frac{\partial\theta}{\partial\phi}\right|_x . \tag{2.30}$$

By application of the chain rule and Equation (2.9), this differential equation may be recast as

$$\frac{D}{DS}\left(\ell n \left.\frac{\partial\phi}{\partial\beta}\right|_x\right) = \frac{1}{r\cos\theta}\left.\frac{\partial\theta}{\partial\phi}\right|_x \quad . \tag{2.31}$$

The integration of this differential equation along a streamline requires that $\left.\frac{\partial\theta}{\partial\phi}\right|_x$ also be known.

The differential equation for $\left.\frac{\partial\theta}{\partial\phi}\right|_x$ may be derived as follows: substitution of the transformation operator in Equation (2.8) into the streamline Equation (2.18) yields

$$\cos\theta\cos\Gamma\frac{\partial\theta}{\partial x} + \frac{\sin\theta}{r}\frac{\partial\theta}{\partial\phi} = -\frac{\sin\theta\sin\Gamma}{r} - \frac{1}{2}\left(\frac{u_\infty}{u_e}\right)^2\left(\frac{1}{h}\frac{\partial C_p}{\partial\beta}\right) \quad . \tag{2.32}$$

Solving for $\frac{\partial\theta}{\partial x}$ gives

$$\frac{\partial\theta}{\partial x} = -\frac{1}{\cos\theta\cos\Gamma}\left\{\frac{\sin\theta}{r}\frac{\partial\theta}{\partial\phi} + \frac{\sin\Gamma\sin\theta}{r} + \frac{1}{2}\left(\frac{u_\infty}{u_e}\right)^2\left(\frac{1}{h}\frac{\partial C_p}{\partial\beta}\right)\right\} \quad . \tag{2.33}$$

Taking the partial derivative with respect to ϕ of Equation (2.32) yields

$$\cos\theta\cos\Gamma\frac{\partial^2\theta}{\partial\phi\partial x} + \frac{\sin\theta}{r}\frac{\partial^2\theta}{\partial\phi^2} - \sin\theta\cos\Gamma\frac{\partial\theta}{\partial x}\frac{\partial\theta}{\partial\phi} + \frac{\cos\theta}{r}\left(\frac{\partial\theta}{\partial\phi}\right)^2 =$$

$$-\frac{\sin\Gamma\cos\theta}{r}\frac{\partial\theta}{\partial\phi} + \frac{1}{2}\left(\frac{u_\infty}{u_e}\right)^2\left(\cos\theta\cos\Gamma\frac{\partial C_p}{\partial x}\frac{\partial\theta}{\partial\phi} + \frac{\sin\theta}{r}\frac{\partial C_p}{\partial\phi}\frac{\partial\theta}{\partial\phi}\right.$$

$$\left. + \sin\theta\cos\Gamma\frac{\partial^2 C_p}{\partial\phi\partial x} - \frac{\cos\theta}{r}\frac{\partial^2 C_p}{\partial\phi^2}\right) - \left(\frac{u_\infty}{u_e}\right)^2\frac{1}{u_e}\frac{\partial u_e}{\partial\phi}\left(\sin\theta\cos\Gamma\frac{\partial C_p}{\partial x}\right.$$

$$\left. - \frac{\cos\theta}{r}\frac{\partial C_p}{\partial\phi}\right) \quad . \tag{2.34}$$

Bernoulli's equation enables the circumferential velocity derivative in this expression to be written as

$$\frac{1}{u_e}\frac{\partial u_e}{\partial \phi} = -\frac{1}{2}\frac{\partial C_p}{\partial \phi}\left(\frac{u_\infty}{u_e}\right)^2 \quad . \tag{2.35}$$

Substitution of this equation and the two transformation operators into Equation (2.34) gives

$$\begin{aligned}\frac{D}{DS}\left(\frac{\partial \theta}{\partial \phi}\right) &= \sin\theta\cos\Gamma\frac{\partial \theta}{\partial x}\frac{\partial \theta}{\partial \phi} - \frac{\cos\theta}{r}\left(\frac{\partial \theta}{\partial \phi}\right)^2 - \frac{\sin\Gamma\cos\theta}{r}\frac{\partial \theta}{\partial \phi} \\ &+ \frac{1}{2}\left(\frac{u_\infty}{u_e}\right)^2\left[\frac{DC_p}{DS}\frac{\partial \theta}{\partial \phi} + \sin\theta\cos\Gamma\frac{\partial^2 C_p}{\partial \phi \partial x} - \frac{\cos\theta}{r}\frac{\partial^2 C_p}{\partial \phi^2}\right] \\ &- \frac{1}{2}\left(\frac{u_\infty}{u_e}\right)^4\frac{\partial C_p}{\partial \phi}\left(\frac{1}{h}\frac{\partial C_p}{\partial \beta}\right) \quad . \end{aligned} \tag{2.36}$$

Substitution of Equation (2.33) into this equation yields

$$\begin{aligned}\frac{D}{DS}\left(\left.\frac{\partial \theta}{\partial \phi}\right|_x\right) &= -\frac{\sin\theta}{\cos\theta}\left(\left.\frac{\partial \theta}{\partial \phi}\right|_x\right)\left[\frac{\sin\theta}{r}\left(\left.\frac{\partial \theta}{\partial \phi}\right|_x\right) + \frac{\sin\Gamma\sin\theta}{r}\right. \\ &\left. + \left(\frac{u_\infty}{u_e}\right)^2\frac{1}{2h}\frac{\partial C_p}{\partial \beta}\right] - \frac{\cos\theta}{r}\left(\left.\frac{\partial \theta}{\partial \phi}\right|_x\right)^2 - \frac{\sin\Gamma\cos\theta}{r}\left(\left.\frac{\partial \theta}{\partial \phi}\right|_x\right) \\ &+ \frac{1}{2}\left(\frac{u_\infty}{u_e}\right)^2\left[\frac{DC_p}{DS}\left(\left.\frac{\partial \theta}{\partial \phi}\right|_x\right) + \sin\theta\cos\Gamma\frac{\partial^2 C_p}{\partial \phi \partial x} - \frac{\cos\theta}{r}\frac{\partial^2 C_p}{\partial \phi^2}\right] \\ &- \frac{1}{2}\left(\frac{u_\infty}{u_e}\right)^4\frac{\partial C_p}{\partial \phi}\left(\frac{1}{h}\frac{\partial C_p}{\partial \beta}\right) \quad . \end{aligned} \tag{2.37}$$

Note that

$$\frac{DC_p}{DS} - \frac{\sin\theta}{\cos\theta}\left(\frac{1}{h}\frac{\partial C_p}{\partial \beta}\right) = \cos\theta\cos\Gamma\frac{\partial C_p}{\partial x} + \frac{\sin\theta}{r}\frac{\partial C_p}{\partial \phi} + \frac{\sin^2\theta\cos\Gamma}{\cos\theta}\frac{\partial C_p}{\partial x}$$

$$- \frac{\sin\theta}{r}\frac{\partial C_p}{\partial \phi} \quad .$$

This may be simplified and written as

$$\frac{DC_p}{DS} - \frac{\sin\theta}{\cos\theta}\left(\frac{1}{h}\frac{\partial C_p}{\partial \beta}\right) = \frac{\cos\Gamma}{\cos\theta}\frac{\partial C_p}{\partial x} \quad .$$

Substitution of this expression into Equation (2.37) leaves

$$\frac{D}{DS}\left(\frac{\partial\theta}{\partial\phi}\right) = -\frac{\sin^2\theta}{r\cos\theta}\left(\frac{\partial\theta}{\partial\phi}\right)^2 - \frac{\sin^2\theta\sin\Gamma}{r\cos\theta}\left(\frac{\partial\theta}{\partial\phi}\right) - \frac{\cos\theta}{r}\left(\frac{\partial\theta}{\partial\phi}\right)^2$$

$$- \frac{\sin\Gamma\cos\theta}{r}\left(\frac{\partial\theta}{\partial\phi}\right) + \frac{1}{2}\left(\frac{u_\infty}{u_e}\right)^2\left(\frac{\partial\theta}{\partial\phi}\right)\frac{\cos\Gamma}{\cos\theta}\frac{\partial C_p}{\partial x} + \frac{1}{2}\left(\frac{u_\infty}{u_e}\right)^2$$

$$\cdot\left[\sin\theta\cos\Gamma\frac{\partial^2 C_p}{\partial\phi\partial x} - \frac{\cos\theta}{r}\frac{\partial^2 C_p}{\partial\phi^2}\right] - \frac{1}{2}\left(\frac{u_\infty}{u_e}\right)^4\frac{\partial C_p}{\partial\phi}\left(\frac{1}{h}\frac{\partial C_p}{\partial\beta}\right) \quad . \qquad (2.38)$$

With the substitution of the transformation operator in (2.7), Equation (2.38) may be rewritten as

$$\frac{D}{DS}\left(\left.\frac{\partial\theta}{\partial\phi}\right|_x\right) = -\frac{\left(\left.\frac{\partial\theta}{\partial\phi}\right|_x\right)\left(\left.\frac{\partial\theta}{\partial\phi}\right|_x + \sin\Gamma\right)}{r\cos\theta} + \frac{1}{2}\left(\frac{u_\infty}{u_e}\right)^2\left[\sin\theta\cos\Gamma\frac{\partial^2 C_p}{\partial\phi\partial x}\right.$$

$$\left. - \frac{\cos\theta}{r}\frac{\partial^2 C_p}{\partial\phi^2}\right] + \frac{1}{2}\left(\left.\frac{\partial\theta}{\partial\phi}\right|_x\right)\left(\frac{u_\infty}{u_e}\right)^2\frac{\cos\Gamma}{\cos\theta}\frac{\partial C_p}{\partial x} + \frac{1}{2}\left(\frac{u_\infty}{u_e}\right)^4\frac{\partial C_p}{\partial\phi}$$

$$\cdot\left[\sin\theta\cos\Gamma\frac{\partial C_p}{\partial x} - \frac{\cos\theta}{r}\frac{\partial C_p}{\partial\phi}\right] \quad . \qquad (2.39)$$

This differential equation involves first and second derivatives of the pressure coefficient which are supplied by either the analytical solution or the spline fit if only experimental pressures are supplied. Note, however, that $\frac{\partial^2 C_p}{\partial x^2}$ does not appear and thus the spline fit in the x-direction is simplified. Equations (2.31) and (2.39) may be integrated to give the scale factor (Equation (2.23)) along an inviscid surface streamline. The differential equations are singular at the stagnation point; therefore, the geometric position where the integration of the streamline equations begins must be some distance away from the stagnation point (Ref. 9). The technique used here depends on whether the inviscid properties are obtained from experimental pressure data or an analytical potential solution.

When experimental pressure data are used, there are generally insufficient data near the stagnation point to adequately determine the pressure distribution needed for the integration of the streamline equations near the nose. A potential panel method, USSAERO, was used to obtain additional pressure data in the nose region. It was found that this pressure data and the experimental pressure data, at the most forward position, were reasonably close to a spherical pressure distribution about the stagnation point. A spherical pressure distribution produces streamlines along spherical meridians from the stagnation point to the sphere-afterbody interface. (See Figure 4.1 on page 35.) On the spherical cap, the streamline geometry and metric are given in Ref. 14. Integration of the streamline differential equations begins at the sphere-afterbody interface. For a given circumferential angle ϕ_0, page 31 of Ref. 9 gives the initial streamline slope as

$$\theta_o = \cos^{-1}\left\{(\cos\alpha_{eff}\cos\Gamma_I - \sin\alpha_{eff}\sin\Gamma_I\cos\phi_o)/\sin\psi_o\right\} \qquad (2.40)$$

where the spherical angle is determined from

$$\psi_o = \cos^{-1}\left\{\cos\alpha_{eff}\sin\Gamma_I + \sin\alpha_{eff}\cos\Gamma_I\cos\phi_o\right\} \quad . \qquad (2.41)$$

The initial value of $(\partial\theta/\partial\phi)_o$ for Equation (2.39) is obtained by differentiating Equation (2.40). The result is

$$\left(\frac{\partial\theta}{\partial\phi}\right)_o = -\frac{\sin\alpha_{eff}\sin\Gamma_I\sin\phi_o - \cos\theta_o\cos\psi_o\left(\frac{\partial\psi}{\partial\phi}\right)_o}{\sin\psi_o\sin\theta_o} \quad . \qquad (2.42)$$

From Ref. 14, the streamline metric on the spherical cap is given by $h = R_{per}\sin\psi$; and since $r = R_{per}\cos\Gamma$, Equation (2.23) gives

$$\left[\left.\frac{\partial\phi}{\partial\beta}\right|_x\right]_o = \frac{\sin\psi_o}{\cos\Gamma_o\cos\theta_o} \qquad (2.43)$$

as the initial value for Equation (2.31) at the sphere-afterbody interface.

When an analytical potential solution is known, the inviscid velocity components can be used to obtain an analytical expression for the streamline angle, θ, and its circumferential derivative, $\partial\theta/\partial x$. Then Equation (2.39) is not needed and Equations (2.9), (2.10), and (2.31) can be integrated numerically to determine the streamline location and metric. Initial conditions for the streamline location are

determined from an axial position and circumferential angle near the stagnation point. As discussed in Ref. 9, the initial value of $(\partial\phi/\partial\beta)_0$, and hence h_0, for Equation (2.31) is arbitrary. Since Equation (2.31) is used to integrate $\ell n(\partial\phi/\partial\beta)_0$ will have no effect on the numerical integration of this differential equation. The actual value calculated for $\partial\phi/\partial\beta$, however, will be relative to the initial value $(\partial\phi/\partial\beta)_0$.

SECTION 3

BOUNDARY LAYER METHODS

The present computer program has the option of employing either Blottner's (Ref. 2) or Hall's (Ref. 4) boundary layer method. For each method a solution is obtained through the use of a finite-difference technique. After application of the respective transformation, the governing equations are then cast in second-order accurate finite-difference form. Since the governing equations are parabolic, the boundary layer may be calculated by "marching" downstream in a step-by-step fashion along an inviscid surface streamline.

Blottner's method involves solving the governing equations written in F-V similarity form. These equations are obtained by the application of the Levy-Lees' transformation defined for incompressible flow as

$$\xi(s) = K\rho\mu\, u_\infty \int_0^s \frac{u_e}{u_\infty} r^2\, ds \tag{3.1}$$

and

$$\eta(s,n) = \frac{u_e\, r\rho n}{\sqrt{2\xi}} \sqrt{K} \quad . \tag{3.2}$$

In the axisymmetric analogue, the body radius r is replaced by the scale factor h and s is distance along the inviscid surface streamline. This transformation creates a (ξ,η) computational grid from the (s,n) physical grid. The computational grid has been effectively stretched in both the normal and tangential directions. The resulting equations are

$$2\xi \frac{\partial F}{\partial \xi} + \frac{\partial V}{\partial \eta} + F = 0 \tag{3.3}$$

and

$$2\xi F \frac{\partial F}{\partial \xi} + V \frac{\partial F}{\partial \eta} + \bar{\beta}(F^2 - 1) = \frac{\partial^2 F}{\partial \eta^2} \tag{3.4}$$

where

$$V = 2\xi \left(F \frac{\partial \eta}{\partial x} + \rho v h / \sqrt{2\xi / K} / K \rho \mu u_e h^2 \right) \tag{3.5}$$

$$F = u/u_e \tag{3.6}$$

and

$$\bar{\beta} = \frac{2\xi}{u_e} \frac{du_e}{d\xi} \tag{3.7}$$

(see Appendix A.1). These equations are then cast in finite difference form using the Crank-Nicholson scheme to yield a system which is second-order accurate in both spatial directions. The resulting system can be conveniently written as

$$A_2 F_{i+1,j-1} + B_2 F_{i+1,j} + C_2 V_{i+\frac{1}{2},j-1} + E_2 V_{i+\frac{1}{2},j} = D_2 \tag{3.8}$$

where

$$A_2 = \Delta\eta(\tfrac{1}{4} + \xi_{i+\frac{1}{2}}/\Delta\xi)$$

$$B_2 = \Delta\eta(\tfrac{1}{4} + \xi_{i+\frac{1}{2}}/\Delta\xi)$$

$$C_2 = -1$$

$$E_2 = 1$$

and

$$D_2 = \Delta\eta(-\tfrac{1}{4} + \xi_{i+\frac{1}{2}}/\Delta\xi)(F_{i,j} + F_{i,j-1})$$

for the continuity equation and

$$A_1 F_{i+1,j-1} + B_1 F_{i+1,j} + C_1 F_{i+1,j+1} + E_1 V_{i+\frac{1}{2},j} = D_1 \qquad (3.9)$$

where

$$A_1 = -\tfrac{1}{2}(1 + \tfrac{1}{2}\Delta\eta\, \bar{V}_{i+\frac{1}{2},j})$$

$$B_1 = 1 + \Delta\eta^2\, \bar{F}_{i+1,j}\,(\bar{\beta}_{i+\frac{1}{2}} + 2\xi_{i+\frac{1}{2}}/\Delta\xi)$$

$$C_1 = -\frac{1}{2}(1 - \frac{1}{2}\Delta\eta\, \bar{V}_{i+\frac{1}{2},j})$$

$$E_1 = \frac{1}{4}\Delta\eta\,(F_{i,j+1} - F_{i,j-1} + \bar{F}_{i+1,j+1} - \bar{F}_{i+1,j-1})$$

and

$$D_1 = \tfrac{1}{2}(F_{i,j+1} - 2F_{i,j} + F_{i,j-1}) + \tfrac{1}{2}\Delta\eta^2\, \bar{\beta}_{i+\frac{1}{2}} \Big[(1 + \bar{F}^2_{i+1,j}) + (1 - F^2_{i,j})\Big] + \frac{\Delta\eta}{4}\, \bar{V}_{i+\frac{1}{2},j} \Big[\bar{F}_{i+1,j+1} - \bar{F}_{i+1,j-1}\Big] + \Delta\eta^2\, \varepsilon_{i+\frac{1}{2}}\,(\bar{F}^2_{i+1,j} + F^2_{i,j})/\Delta\xi$$

for the momentum equation (see Appendix A.2). To provide that only a linear system of equations needs be solved to obtain F, the nonlinear terms in the finite difference expressions have been linearized using the Newton-Raphson technique. This will necessitate repeated iteration

in order to achieve a converged solution to the actual nonlinear equations. The bar indicates quantities from a previous iteration.

By virtue of the transformation in Equations (3.1) and (3.2), Blottner's method may be applied at the stagnation point to yield a limiting velocity profile. In addition, a similarity solution is possible since $\xi = 0$ at that point (see Appendix A.3). The boundary conditions consist of the edge and wall conditions. At the boundary-layer edge the condition

$$F(\eta = \eta_e) = 1$$

is applied at each step of the integration. The value of the normal coordinate at the boundary layer edge, η_e, must be provided initially and must be large enough to account for the entire boundary layer thickness. The no-slip condition demands that

$$F(\eta = 0) = 0$$

at each step. The pressure gradient parameter, $\bar{\beta}$, is related to the velocity gradient and for spherical flow, $\bar{\beta}$, becomes 1/2.

The system of equations may then be solved to yield the limit of F at the stagnation point through use of the modified Davis algorithm (Ref. 2). This algorithm solves a coupled system of equations. The profile slope at the wall may be expressed by a second-order accurate expression which in turn is used to evaluate $(C_f \sqrt{Re_L})_\omega$.

At points away from the stagnation region it is necessary to solve the complete, nonsimilar system of equations (Equations (3.8) and (3.9)) which involve the transformed step size, $\Delta\xi$, along a streamline.

With a prescribed step size along a streamline, the expression for ξ may be numerically integrated for subsequent use in the boundary layer equations. The calculation of the pressure gradient parameter, $\overline{\beta}$, which is defined as

$$\overline{\beta} = \frac{2\xi}{u_e} \frac{du_e}{d\xi} \quad ,$$

is evaluated at the midpoint of the computational interval along the body surface. On the nose region of a spherically capped body, this term reduces to

$$\overline{\beta}_{i+\frac{1}{2}} = \frac{2}{R_{per}} \left(\frac{u_\infty}{u_e}\right)^2 \frac{\xi_{i+\frac{1}{2}}}{h^2} \left.\frac{du_e}{d\psi}\right|_{i+\frac{1}{2}}$$

and on the afterbody it becomes

$$\overline{\beta}_{i+\frac{1}{2}} = \frac{2\xi_{i+\frac{1}{2}}}{h^2} \left(\frac{u_\infty}{u_e}\right)^2 \left[\frac{\partial u_e}{\partial x} \frac{DX}{DS} + \frac{\partial u_e}{\partial \phi} \frac{D\phi}{DS}\right]_{i+\frac{1}{2}} .$$

The derivative $\frac{\partial u_e}{\partial \psi}$ and both $\frac{\partial u_e}{\partial x}$ and $\frac{\partial u_e}{\partial \phi}$ are supplied by subroutines SPHCAP and INVISD, respectively. The two total derivatives along a streamline, $\frac{DX}{DS}$ and $\frac{D\phi}{DS}$, are used in conjunction with the streamline integration (see INVISCID SURFACE STREAMLINES). The nonsimilar equations may then be solved for F using the same computational technique as was used at the stagnation point.

Hall's method involves solving the governing equations written in terms of dimensionless primitive variables. Hall employs the customary transformation

$$s^* = s/L$$

$$n^* = \frac{n}{L}\sqrt{R_{e_L}}$$

$$r^* = r/L$$

$$u^* = u/u_\infty$$

$$v^* = v/u_\infty$$

where

$$\sqrt{R_{e_L}} = \sqrt{\rho \frac{u_\infty L}{\mu}}$$

which yields

$$\frac{\partial(u^* r^*)}{\partial s^*} + r^* \frac{\partial v^*}{\partial n^*} = 0 \qquad (3.10)$$

for the continuity equation and

$$u^* \frac{\partial u^*}{\partial s^*} + v^* \frac{\partial u^*}{\partial n^*} = \frac{1}{2} \frac{d(u_e^*)^2}{ds^*} + \frac{\partial^2 u^*}{\partial n^{*2}} \qquad (3.11)$$

for the s-momentum equation (see Appendix A.4). The desired unknown is u* which is the dimensional velocity normalized by the freestream velocity. These equations are then cast in second-order accurate finite-difference form (see Appendix A.5). The resulting system may be expressed as

$$A_2\, u_{i+1,j} + B_2\, u_{i+1,j-1} + C_2\, v_{i+\frac{1}{2},j} + E_2\, v_{i+\frac{1}{2},j-1} = D_2 \qquad (3.12)$$

where

$$A_2 = \frac{\Delta nh_{i+1}}{2\Delta sh_{i+\frac{1}{2}}}$$

$$B_2 = \frac{\Delta nh_{i+1}}{2\Delta sh_{i+\frac{1}{2}}}$$

$$C_2 = 1$$

$$E_2 = -1$$

and

$$D_2 = \frac{\Delta nh_i}{2\Delta sh_{i+\frac{1}{2}}} (u_{i,j} + u_{i,j-1})$$

for the continuity equation and

$$A_1 u_{i+1,j-1} + B_1 u_{i+1,j} + C_1 u_{i+1,j+1} + E_1 v_{i+\frac{1}{2},j} = D_1 \qquad (3.13)$$

where

$$A_1 = -\frac{\bar{v}_{i+\frac{1}{2},j}}{4\Delta n} - \frac{1}{2\Delta n^2}$$

$$B_1 = \frac{\bar{u}_{i+1,j}}{\Delta s} + \frac{1}{\Delta n^2}$$

$$C_1 = \frac{\bar{v}_{i+\frac{1}{2},j}}{4\Delta n} - \frac{1}{2\Delta n^2}$$

$$E_1 = (\bar{u}_{i+1,j+1} - \bar{u}_{i+1,j-1} + u_{i,j+1} - u_{i,j-1})/4\Delta n$$

and

$$D_1 = \frac{\bar{u}^2_{i+1,j} + u^2_{i,j}}{2\Delta s} + \bar{v}_{i+\frac{1}{2},j} \frac{(\bar{u}_{i+1,j+1} - \bar{u}_{i+1,j-1})}{4\Delta n}$$

$$+ \frac{u^2_{e_{i+1}} - u^2_{e_i}}{2\Delta s} + \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{2\Delta n^2}$$

for the momentum equation (note that the stars have been omitted for clarity). These equations have also been linearized using the Newton-Raphson method. The system of equations which results at each station along the body is block tridiagonal in form and may be easily and efficiently solved in the same manner as was used for Blottner's method.

The boundary conditions required for a solution to Hall's equations are that

$$u(s,n_e) = u_e$$

and

$$u(s,o) = 0$$

for the no-slip condition.

To begin the integration of the boundary layer, an initial profile must be known. The stagnation point is an ideal place to start the integration. Hall (Ref. 4) and Geissler (Ref. 10) both utilize the well-known three-dimensional stagnation point boundary-layer solution of Howarth (Ref. 11). This is an unwarranted complication since at the stagnation point

$$u(o,n) = 0$$

at all points across the boundary layer. This may be used as the initial velocity profile in Hall's method, and then the first station away from the stagnation point, along an inviscid surface streamline, may be calculated by the method given above.

3.1. Convergence Criteria

Since the finite-difference equations have been linearized, repeated iteration is necessary in order to obtain a solution to the nonlinear equations. The iterative process could be made to continue until the solution becomes exact (within the accuracy of the computer) but this is no doubt unwarranted. In practical applications, the iterative process is usually allowed to continue until the solution is changing by less than a prescribed amount between successive iterations. This is one definition of a converged solution.

Because the skin friction is the one of the more important parameters of interest, it appears logical that convergence should be based on it. In practical applications, the iterative process should stop when the skin friction changes by less than a prescribed amount between successive iterations. This is the definition most commonly applied in two-dimensional boundary layer cases.

The computational method developed has the option of employing either of these definitions. The input parameter, NC, corresponds to the method which is used to define a converged solution. The option corresponding to NC = 0 specifies that convergence is based on $\left(C_f \ R_{e_L}\right)_e$ changing by less than 0.5 percent between successive iterations. Covergence is based on the velocity at each grid point changing by

less than 0.1 percent between iterations when NC = 1. Table 3.1 illustrates the effect of this option on the separation point for a sphere in incompressible flow.

Table 3.1. Effect of Convergence Criteria on Boundary-Layer Separation for Hall's and Blottner's Methods on Sphere in Incompressible Flow

Method	NC	Separation Angle (Deg.)	Steps to Separation
Hall	0	107.43	184
$\Delta s = 0.01$ $\Delta n = 0.0471$	1	104.83	184
Blottner	0	105.75	278
$\Delta \xi = 0.005$ $\Delta \eta = 0.11539$	1	104.94	276

40 Points Across Boundary Layer Initially

3.2. Boundary Layer Edge Criteria

Since the velocity in the boundary layer only approaches the value of the inviscid stream asymptotically, an effective edge must be imposed. The velocity at the grid point which is arbitrarily said to lie at the edge is assigned the velocity of the inviscid stream. The relationship of the velocities at the grid points in the region near the imposed edge may then be used to assess whether the actual boundary-layer thickness has been adequately accounted for. According to the classical definition, the boundary layer thickness is adequately represented if the velocity at the point adjacent to the imposed edge is a certain percentage of the velocity at the imposed edge. This percentage is

usually in the range of 99.5 to 99.995 percent. Wang (Ref. 5) employs this definition in his fully three-dimensional technique. A second test for the boundary-layer edge could be constructed which utilizes the friction parameter, $\left(C_f \sqrt{R_{e_L}}\right)_e$, at the edge of the boundary layer as the governing criterion. If this parameter is below a prescribed limit, the imposed edge may be considered to adequately account for the boundary-layer thickness.

Both of the tests described above are included as options in the present computational method. The parameter NT specifies an option to be used for the edge test. The option NT = 0 specifies that the edge test be based on the classical definition in which the tolerance is 99.95 percent. The second test, which corresponds to NT = 1, requires that $\left(C_f \sqrt{R_{e_L}}\right)_e$ be less than 0.005 for the boundary-layer thickness to be adequate. Table 3.2 illustrates the effect of both the edge test and convergence test options on the separation point for a sphere in incompressible flow. Note that the more stringent option, NT = 1, results in the addition of points at the boundary-layer edge and a more accurate separation point. The results generated in conjunction with the option corresponding to NT = 0 could most likely be improved if the respective tolerance were to be decreased. It is evident from the results obtained with Hall's method that the boundary layer is undoubtedly thickening. Since Hall makes use of primitive variables, a growing boundary layer will require that the outer edge be adjusted occasionally. The transformed normal coordinate used in Blottner's method has provisions to account for the growth of the boundary layer. Because of this, it is seldom necessary to manually shift the outer edge.

Table 3.2. Effect of Edge Test and Convergence Test on Separation for Hall's and Blottner's Methods on Sphere in Incompressible Flow

Method	NC	NT	Separation Angle (Deg.)	Steps to Separation	Points Added
Hall	0	0	108.59	194	23
$\Delta s = 0.01$ $\Delta n = 0.471055$	0	1	107.43	184	31
	1	0	104.89	184	18
	1	1	104.83	184	30
Blottner	0	0	105.02	278	0
$\Delta\xi = 0.005$ $\Delta\eta = 0.11539$	0	1	105.75	278	5
	1	0	104.94	276	0
	1	1	104.94	276	4

Initially 40 Points Across Boundary Layer

SECTION 4

SURFACE PRESSURE DISTRIBUTION

If an analytical potential solution is not available for a particular configuration, experimental pressure data must be applied. The accurate surface fitting of the pressure data is critical not only to the calculation of the boundary-layer properties but also to the calculation of the inviscid surface streamlines. Near the nose region of a body where pressure gradients are relatively large, experimental pressure data are generally not available. The region of the body downstream of the nose generally experiences more moderate pressure gradients and sufficient experimental data are provided to model a surface pressure distribution. After investigating several methods for surface fitting experimental pressure data, it was found that a doubly quadratic spline would adequately model the pressure distribution downstream of the nose.

As mentioned earlier, a potential panel method, USSAERO, was used to calculate additional pressure data in the nose region. Attempts to use a doubly quadratic spline to blend the pressures calculated by the USSAERO code in the nose region with the experimental data downstream were unsucessful. To model the pressure distribution in the nose region, an alternate approach was employed. This method has been tested for a sphere-ogive-cylinder only. First, the pressure data calculated by the USSAERO code were blended with the experimental data at the most forward station and then they are plotted as a function of the spherical angle ψ about the stagnation point (see Figure 5.2 on page 42). It was found that the Fourier cosine series

$$C_p(\psi) = A_o + \sum_{n=1}^{9} A_n \cos(n\psi) \qquad (4.1)$$

represented the pressure distribution in the nose region quite satisfactorily. In this series, the spherical angle ψ is given by

$$\psi = \cos^{-1}\left\{\cos \alpha_{eff} \sin \Gamma + \sin \alpha_{eff} \cos \Gamma \cos \phi\right\} \qquad (4.2)$$

where α_{eff} is the angle between the body axis of symmetry and the line which passes through the stagnation point (see Figure 4.1). Note that $\psi = 0$ corresponds to the stagnation point.

The coefficients in the Fourier series, A_n $(n = 0,9)$, are obtained from the solution of ten simultaneous equations generated from the application of Equation (4.1) to ten distinct points on a curve faired through the pressure data calculated from USSAERO. One point must be the stagnation point itself which is determined by interpolating data from USSAERO.

Away from the nose region, pressure gradients usually become smaller and a doubly quadratic spline may be used to fit the experimental pressure data downstream of the interface. In order to describe the doubly quadratic spline, consider the singly quadratic spline first. An interval

$$x_1 \leq x \leq x_N$$

is divided into N subintervals. Each interior subinterval rage is

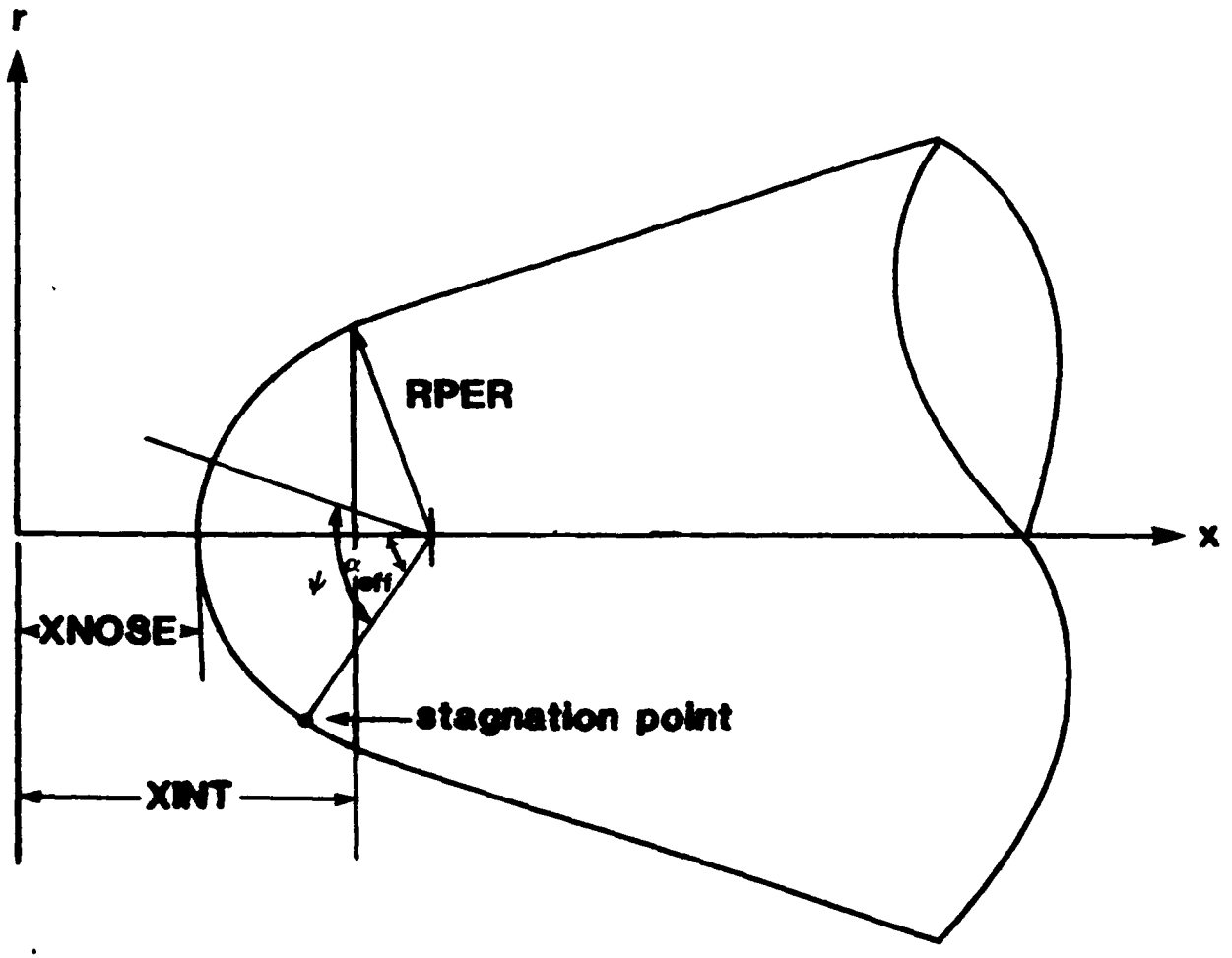


Figure 4.1. Geometric Parameters

$$\frac{x_{n-1} + x_n}{2} \leq x \leq \frac{x_n + x_{n+1}}{2} \quad (n = 2, N - 1)$$

and the subintervals on the left and right boundaries range from

$$x_1 \leq x \leq \frac{x_1 + x_2}{2}$$

and

$$\frac{x_{N-1} + x_N}{2} \leq x \leq x_N \quad ,$$

respectively. The dependent variable at each of the points x_n is designated by y_n (see Figure 4.2).

There exist N-1 midpoints in the total interval. The midpoints, denoted by X_n, may be computed by the above relations. Corresponding to each of the midpoints X_n there is a yet undetermined dependent variable Y_n. Each of the Y_n's is determined such that there is continuity of the function and its first derivative between adjacent subintervals. The first derivative must be specified on the left and right boundaries of the interval. This will yield a system of N-1 linear equations for Y_n (n=1,N-1). This system may be expressed as

$$(b_1 + c_1)\, Y_1 + d_1 Y_2 = \frac{\left.\frac{dy}{dx}\right|_{x_1}}{2} + y_1 b_1 + y_2\,(c_1 + d_1)$$

$$a_n Y_{n-1} + (b_n + c_n) Y_n + d_n Y_{n+1} = y_n\,(a_n + b_n) + y_{n+1}\,(d_n + c_n) \quad (n=2,N-2)$$

$$a_{N-1} Y_{N-2} + (b_{N-1} + c_{N-1})\, Y_{N-1} = \frac{\left.\frac{dy}{dx}\right|_{x_N}}{2} + y_{N-1}\,(a_{N-1} + b_{N-1}) + y_N (c_{N-1})$$

(4.3)

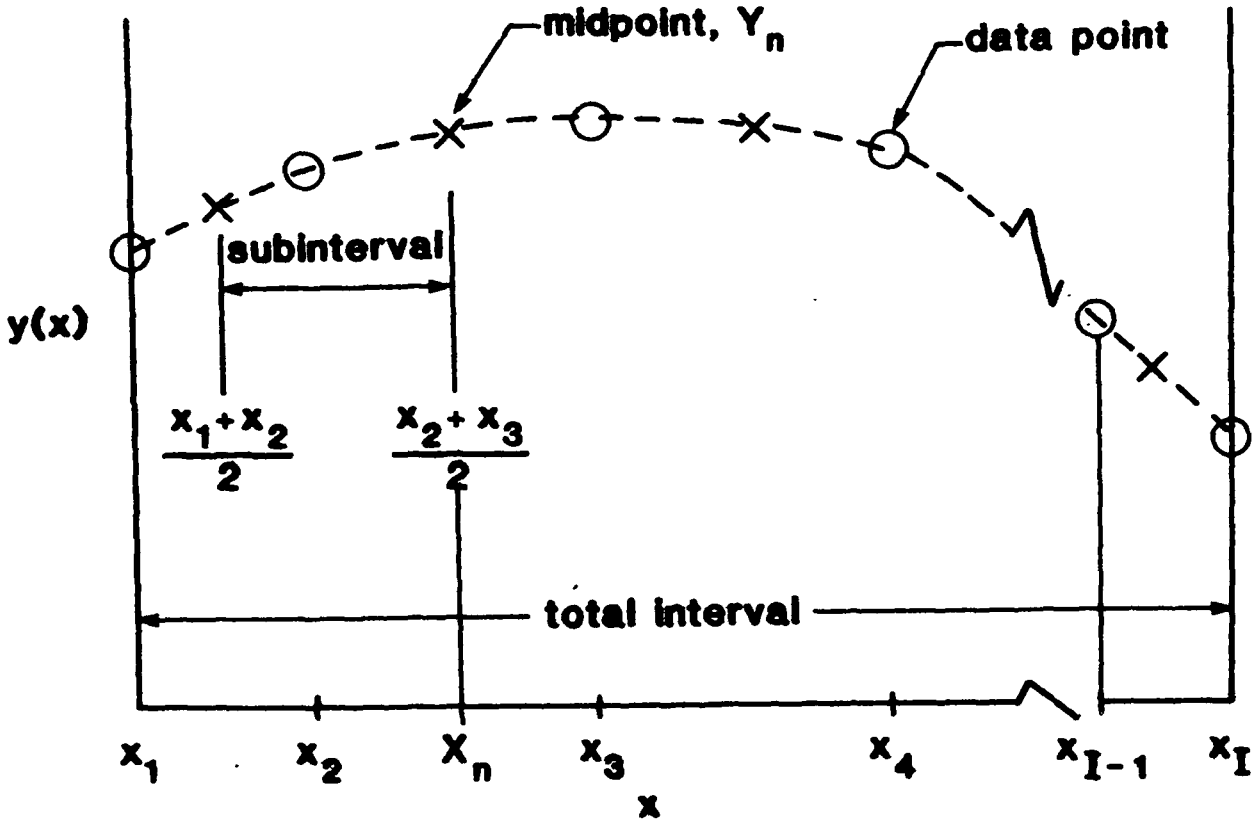


Figure 4.2. Quadratic Spline Parameters

where

$$b_1 = 2/\Delta x_2$$

$$a_n = \Delta x_{n+1}/[x_n(\Delta x_{n+1} + \Delta x_n)]$$

$$b_n = [2 + \Delta x_n/\Delta x_{n+1}]/[\Delta x_{n+1} + \Delta x_n]$$

$$c_n = [2 + \Delta x_{n+2}/\Delta x_{n+1}]/[\Delta x_{n+2} + \Delta x_{n+1}]$$

$$d_n = x_{n+1}/[\Delta x_{n+2}(\Delta x_{n+2} + \Delta x_{n+1})]$$

$$c_{N-1} = 2/\Delta x_n$$

and

$$\Delta x_n = x_n - x_{n-1} \quad .$$

This system forms a tridiagonal matrix and the unknowns may be obtained through use of the Davis algorithm (Ref. 2).

A second-order polynomial about the point x_n which may be written as

$$y(x) = y_n + y_n'(x - x_n) + y_n'' \frac{(x - x_n)^2}{2} \qquad (4.4)$$

is applied at the single data point which lies within each subinterval. This equation contains only two unknowns since y_n is known at the data point x_n. On a given interior subinterval, y_n' and y_n'' are related to the dependent variables Y_{n-1} and Y_n (which have already been determined) and can be expressed as

$$y'_n = \frac{2}{\Delta x_{n+1} + \Delta x_n} \left[\Delta x_n \left(\frac{Y_n - y_n}{\Delta x_{n+1}} \right) - \Delta x_{n+1} \left[\frac{Y_{n-1} - y_n}{\Delta x_n} \right] \right] \tag{4.5}$$

and

$$y''_n = \frac{8}{\Delta x_{n+1} + \Delta x_n} \left[\frac{Y_n - y_n}{\Delta x_{n+1}} + \frac{Y_{n-1} - y_n}{\Delta x_n} \right] . \tag{4.6}$$

Thus to determine the value of the dependent variable, y, and its derivatives at any position on the total interval, all that need be done is to determine in which subinterval the independent variable lies. The corresponding coefficients in the quadratic expression (Equation (4.4)) may be generated from Equations (4.5) and (4.6).

The extension of the one-dimensional quadratic spline to two dimensions is a relatively simple process and is performed as follows. One-dimensional quadratic splines $y(x,\phi_k)$ are formed for specified values of ϕ_k. For a given value of x, $y(x,\phi_k)$ and $\frac{\partial y}{\partial x}(x,\phi_k)$ are calculated for each ϕ_k. These values are then fitted by a quadratic spline in the ϕ direction. These splines can then be used to calculate $y(x,\phi)$, $\frac{\partial y}{\partial x}(x,\phi)$ and $\frac{\partial y}{\partial \phi}(x,\phi)$ for a given value ϕ.

The quadratic spline yields a function which is continuous and has a continuous first derivative. The second derivative is continuous (and constant) over each subinterval, but is not constrained to be continuous at the junction of the subintervals. It is possible for inflection points to occur only at these junctions. Should an inflection point be desired at a specific location, it may be included simply by the addition of two data points such that the midpoint of this interval becomes an endpoint of a subinterval.

SECTION 5

STREAMLINES ON SPHERICALLY CAPPED GEOMETRIES

Experimental pressure data were obtained on a sphere-ogive-cylinder at $\alpha = 45°$ from the first row of pressure taps to the base region. No experimental pressure data were obtained on the spherical cap. To assist in modeling the pressure distribution on the spherical cap, the USSAERO potential code was used to calculate pressure data and these data were interpolated to locate the stagnation point. Due to the large pressure variation over the nose region, the doubly quadratic spline function used to model the pressure distribution downstream of the sphere-ogive interface was found to be unsatisfactory for the nose region. An alternate approach described in SECTION 4 was to graph the calculated pressure data from USSAERO on the spherical cap as a function of the angle ψ, given by Equation (4.2), which is the spherical angle measured about an axis passing through the stagnation point and the center of the sphere (see Figure 4.1). The results are given on Figure 5.2 and they show that the pressure distribution is reasonably close to a spherically symmetric one. With the assumption of a spherically symmetric pressure distribution, the streamlines on the spherical cap will simply follow spherical meridians about the axis through the stagnation point and the center of the sphere. The angle between this axis and the body axis is α_{eff} in Figure 4.1 which is quite different from the actual angle of attack, α.

For spherical flow, the pressure distribution along one meridian is indistinguishable from another and the boundary layer is truly axisymmetric. Note, however, that the magnitude of the pressure over

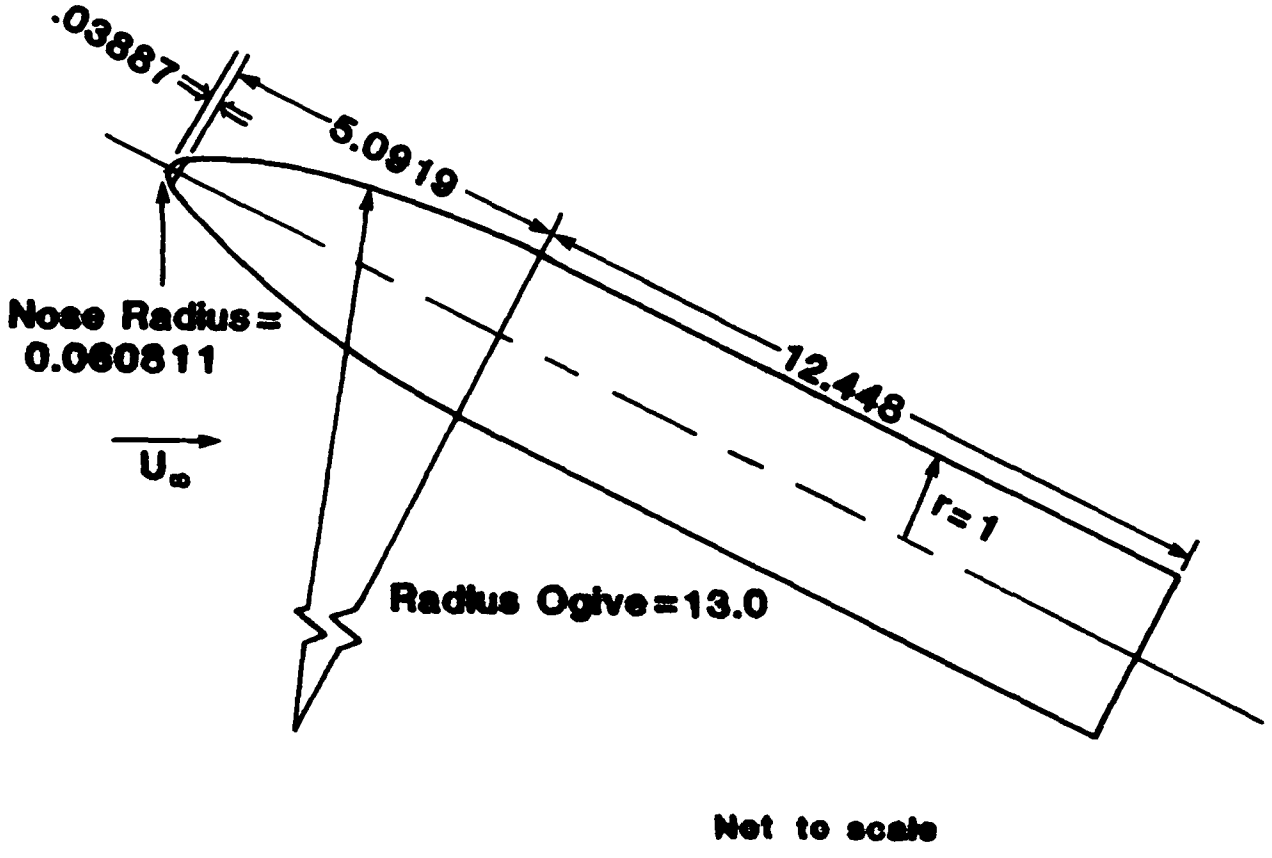


Figure 5.1. Sphere-Ogive-Cylinder Configuration

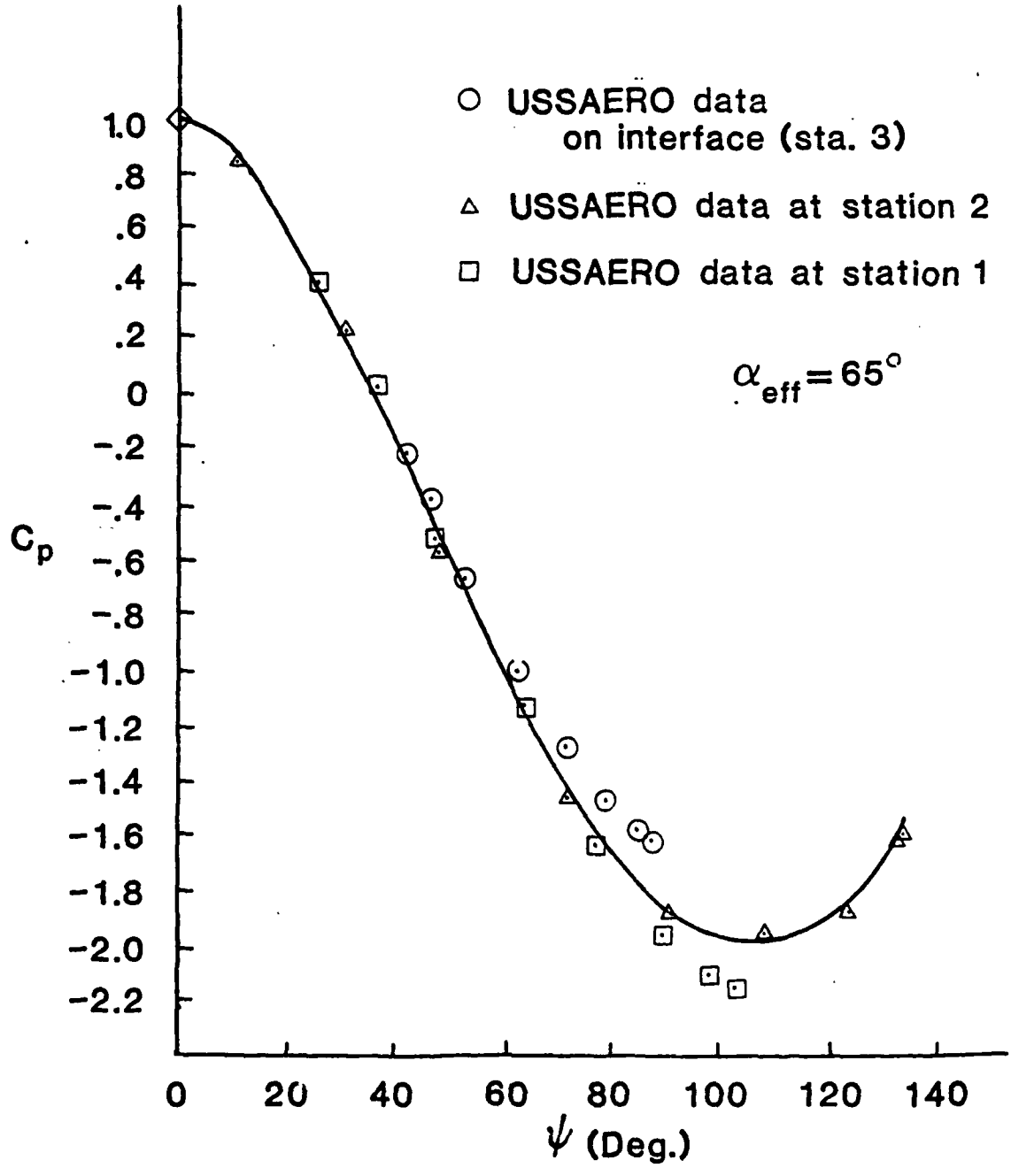


Figure 5.2. Spherical Symmetry of Nose Pressure Distribution

this portion of a sphere is not the same as that over a sphere alone in incompressible flow. The integration of the boundary layer equations continues from the stagnation point along a meridian until the sphere-afterbody interface has been reached. This stopping point is designated by the angle ψ attaining a particular maximum value. This maximum value is a function of the circumferential position on the interface and can be calculated from Equation (4.2). Beyond the interface, the inviscid surface streamlines must be integrated numerically.

SECTION 6

DESCRIPTION OF COMPUTATIONAL METHOD

The method presented traces inviscid surface streamlines while simultaneously computing the properties of the boundary layer up to the separation point. Tracing a streamline involves the numerical integration in a step-by-step fashion of Equations (2.9), (2.10), and (2.18) to determine the axial position, circumferential angle, and the streamline angle (see INVISCID SURFACE STREAMLINES). In conjunction with the differential equations for the streamlines, Equations (2.31) and (2.37) are also integrated to give the scale factor along the streamlines. After each integration increment along a streamline, the boundary-layer equations are then integrated by either Hall's or Blottner's method to determine the local velocity profile across the boundary layer (see BOUNDARY LAYER METHODS). This profile is used to determine the local value of $\left(C_f\sqrt{Re_L}\right)_\omega$. The separation point for the flow along a particular streamline is assumed to occur when this parameter passes through zero. All calculations stop at this point since both the streamline and boundary-layer equations are invalid in the separated region. Several streamlines are calculated to get a distribution around the body.

In order to begin the boundary layer integration, it is first necessary to establish the initial boundary layer velocity profile at the stagnation point. For Blottner's method this neccesitates solving the similar F-V equations while for Hall's method, each point in the

profile is identically zero. For instances in which the pressure distribution is expressed analytically, the integration of the boundary layer continues along an inviscid surface streamline in increments of Δs (which has units of the input geometry). For cases in which only experimental pressures are available, the angle ψ is first calculated given a position on the sphere-afterbody interface. The integration of the boundary layer then proceeds in a step-by-step fashion in increments of the angle ψ, $\Delta\psi$, on the spherical cap until the value of ψ at the interface has been reached. The boundary condition on the fluid velocity at the edge of the boundary layer is a function of ψ only and is obtained at each step during the integration from subroutine SPHCAP. The integration of the streamlines begins at the sphere-afterbody interface.

The boundary-layer profile convergence test is then applied after each iteration of the solution. This computer program employs two options with which to define a converged solution. One option requires that the skin friction parameter, $\left(C_f\sqrt{R_{e_L}}\right)_\omega$, change by less than 0.5 percent between successive iterations in order for the solution to be considered to have converged. The other option requires that the velocity at each grid location change by less than 0.1 percent from the previous iteration.

Once the solution has converged, the edge test is performed. This test effectively determines whether the point at the edge of the boundary layer spans the total thickness. There are two options regarding the edge test. One option requires that the velocity at the grid location just inside the imposed boundary-layer edge be at least 99.95 percent

of the velocity at the boundary in order for the imposed thickness to be adequate. The second option requires that the skin friction parameter, $\left(C_f \sqrt{R_{e_L}}\right)$, at the edge be less than or equal to 0.005. If the option employed should fail, an additional point is added at the boundary-layer edge of both the present and previous computational station. The velocity at the outer grid point of the previous station is assigned the value of the edge velocity at that station. With the addition of the point at the edge, the calculations for the present station are repeated. This procedure is followed until both tests have been satisfied. If the number of points added at the edge should eventually exceed 50, the step size in the normal direction is doubled and every other point within the boundary layer is discarded. At this time the step size along the streamline is also doubled.

Three methods are available with which to integrate the inviscid streamline differential equations. The predictor-corrector method of Milnes (Ref. 12) features rapid execution and has been incorporated into the computational code. This method, however, is not self-starting and makes no check for truncation error (see SUBROUTINE MILNES). The method of Gear is used to generate the starting values. Gear's method (Ref. 12) is useful for instances in which a stiff system of first-order differential equations is being integrated (see SUBROUTINE DGEAR). The last method is the fourth-order Runge-Kutta method. This method is not as well suited to stiff systems because the step size becomes prohibitively small in the attempt to minimize the truncation error during integration (see FUNCTION KRUNGE).

The boundary layer is integrated in increments of Δs along a streamline. The boundary condition on the fluid velocity at the outer edge is obtained from subroutine INVISD. If experimental pressures are supplied, a second subroutine, SPHCAP, provides the necessary conditions for points on the spherical cap. The integration of the boundary layer continues up to the point at which $\left(C_f \sqrt{R_{e_L}}\right)_\omega$ reaches or passes through zero.

At larger angles-of-attack, the streamlines quite frequently wrap around the body so rapidly that it is difficult to resolve the boundary layer at points further down the body. A technique which employs a shift from the windward streamline may be implemented in order to accomplish this. The integration of the boundary layer continues along the windward streamline to the input axial position, XMAX. At this point the circumferential angle is changed from zero to one degree. From this point on the integration continues along this newly defined streamline to the separation point. With this technique it is possible to trace streamlines that otherwise would have been unobtainable.

The separation point along a streamline is approximated by linear interpolation using the last two converged solutions since the boundary-layer profile frequently fails to converge once in the separated region. This entire procedure is repeated for each of the streamlines. The total number of streamlines is an input parameter called KBM.

SECTION 7

RESULTS AND DISCUSSION

In order to illustrate the validity of the techniques employed in the computer program, results are presented for a sphere, ellipsoid of revolution at an angle of attack and a sphere-ogive-cylinder configuration at an angle of attack. Each test case represents a step up in the complexity of the analysis. The results for each geometry consist primarily of a comparison between the solutions obtained by both Hall's and Blottner's methods. Additional results generated for the ellipsoid of revolution at two different angles of attack are compared to fully three-dimensional boundary-layer calculations. All computations were performed on the IBM 370/165 digital computer at North Carolina State University. Computer times in this section are in CPU seconds. All cases start with 40 points across the boundary layer.

7.1. Sphere

The sphere geometry provides the opportunity to validate the computational code itself. The comparisons presented in Tables 3.1, 3.2, and 7.1 serve as verification. Results generaged on both a cylinder and a flat plate compared quite well with the accepted values.

7.2. Ellipsoid of Revolution

The ellipsoid selected for this case had a thickness ratio of 1/4, a total length L = 2a, and was examined at both 12° and 30° angle-of-attack (see Figure 7.1). The potential solution (Ref. 13) was available in the form of an analytical expression (see Appendix A.6). For this case, only the differential equations in (2.9), (2.10) and (2 31) were

Table 7.1. Effect of Convergence and Edge Criteria on Skin Friction for Sphere in Incompressible Flow

	Hall $\Delta s = 0.05 \quad \Delta n = .0471$		$\left(C_f\sqrt{Re_L}\right)_\omega$ Blottner $\Delta s = 0.05 \quad \Delta\eta = 0.11539$	
ψ	NC = 0	NC = 1	NC = 0	NC = 1
20.	2.578	2.579	2.710	2.573
40.	2.430	2.418	2.431	2.417
60	2.150	2.149	2.150	2.149
80	1.656	1.656	1.654	1.657
90	1.249	1.276	1.271	1.271
100	0.650	0.645	0.640	0.642
104	0.202	0.193	0.226	0.226

integrated since analytical expressions for the streamline angle, Equation (2.18), and its circumferential derivative, Equation (2.39), were available (see Appendix A.7). For these cases, $\Delta s = 0.05$ and $\Delta n = 0.0471$ were used for Hall's method, and $\Delta s = 0.05$ and $\Delta\eta = 0.115385$ for Blottner's method.

The results of both Hall's and Blottner's methods are presented for a variety of streamlines on this configuration in Table 7.2. This table includes results that were obtained from the streamline shifting technique described in the section of this thesis labeled DESCRIPTION OF COMPUTATIONAL METHOD. XMAX is the axial position at which a streamline shift from a circumferential position of zero to one degree was made. The results of both methods agree quite well despite the fact that the step size along a streamline differs between the two methods

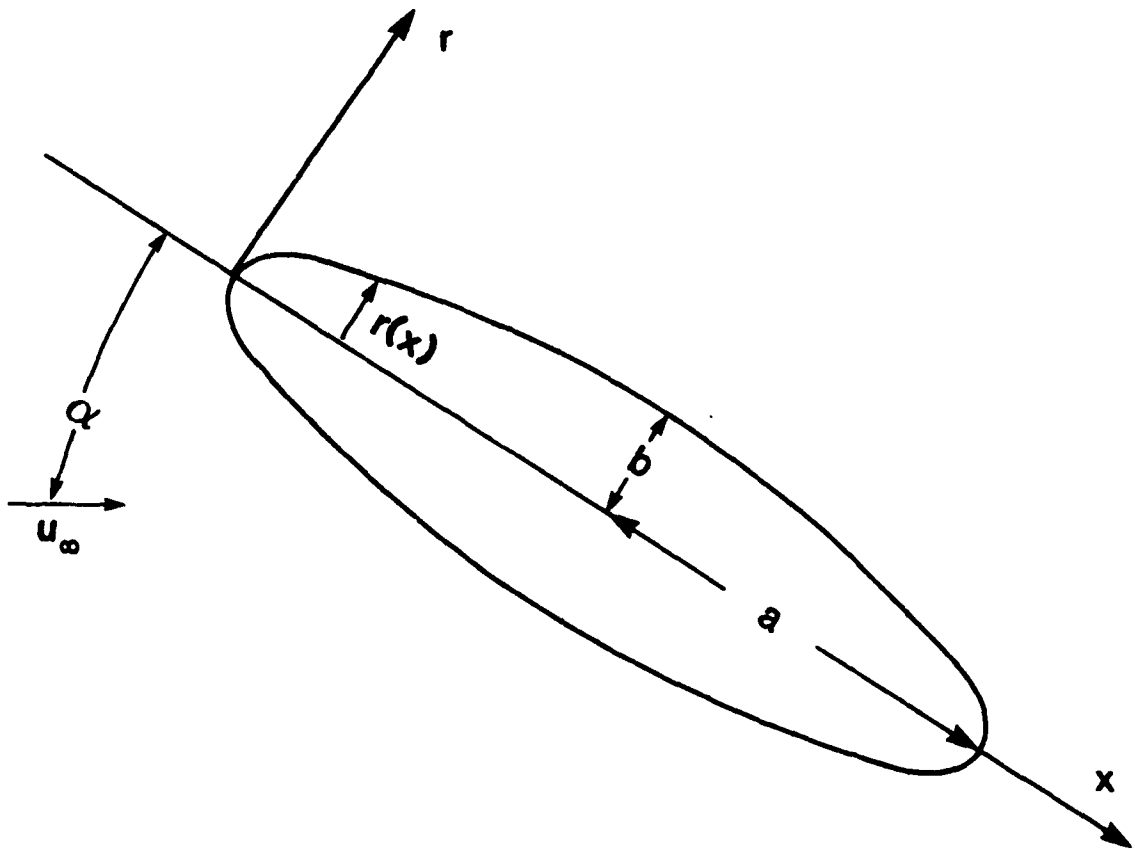


Figure 7.1. Ellipsoid of Revolution

Table 7.2. Comparison of Separation Points Between Hall's and Blottner's Methods on Ellipsoid of Revolution with Thickness Ratio 1/4 at 30° Angle-of-Attack

Method	Beta	XMAX/L	Points Added	Computer Time (sec)	Separation X/L	Separation ϕ
Hall	20.0	--	52	6	0.209	143.28
	50.0	--	55	7	0.222	142.76
	0.0	0.30	53	13	1.363	104.42
	0.0	0.50	49	13	1.626	95.89
	0.0	0.80	48	14	1.857	82.18
Blottner	20.0	--	3	7	0.212	143.94
	50.0	--	3	7	0.220	142.43
	0.0	0.30	1	12	1.366	104.15
	0.0	0.50	2	12	1.640	97.46
	0.0	0.80	0	12	1.857	82.16

by a factor of 20. Hall's method generally required a greater amount of computational time. This is most likely due to the greater number of points that had to be added at the imposed boundary layer edge.

Tables 7.3 and 7.4 compare results generated by the axisymmetric analogue using Hall's and Blottner's methods to the three-dimensional boundary layer calculations of Wang (Ref. 13). In both cases the comparison is increasingly degraded as the leeside of the body is approached. In this instance, both tables suggest that the axisymmetric analogue yields quite good results on the windside of the body. The separated region for this case is shown graphically in Figure 7.2. The separated region as calculated by Wang is also included for comparison.

Table 7.3. Comparison of Separation Points Between Hall's Method and Three-Dimensional Boundary Layer Calculations on Ellipsoid of Revolution with Thickness Ratio 1/4 at 30° Angle-of-Attack

	Circumferential Separation, ϕ, (Degrees)	
Axial Station (X/L)	Hall's Method	3-D Results
0.209	143.28	131.25
0.222	142.76	130.00
0.328	132.56	125.50
1.363	104.42	102.50
1.626	95.89	95.00
1.857	82.18	82.50

Table 7.4. Comparison of Separation Points Between Blottner's Method and Three-Dimensional Boundary Layer Calculations on Ellipsoid of Revolution with Thickness Ratio 1/4 at 30° Angle-of-Attack

	Circumferential Separation, ϕ, (Degrees)	
Axial Station (X/L)	Blottner's Method	3-D Results
0.212	143.94	131.25
0.220	142.43	130.00
1.366	104.15	102.50
1.640	97.46	94.50
1.857	82.16	82.50

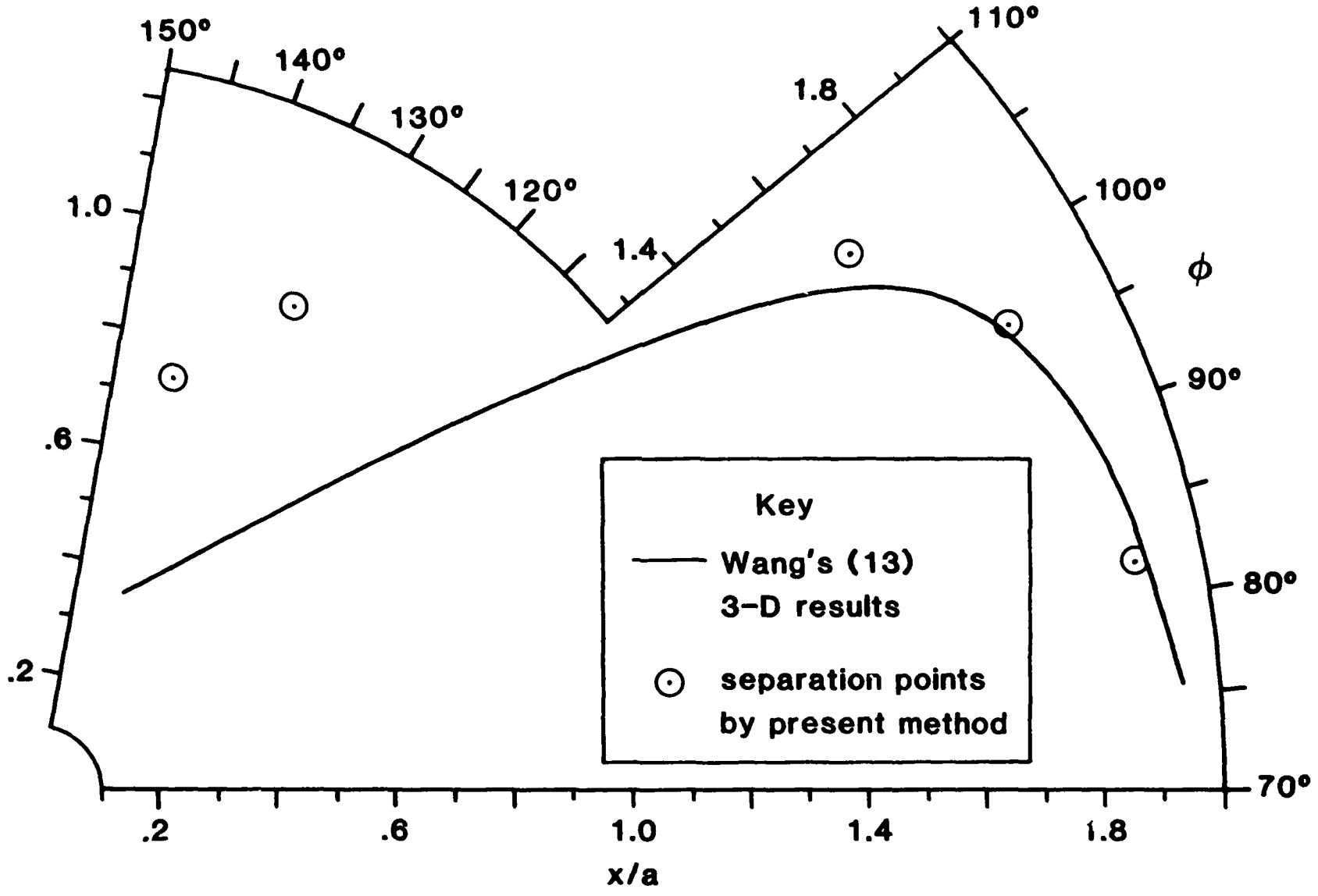


Figure 7.2. Separation Region on Ellipsoid of Revolution with Thickness Ratio 1/4 at 30 Degrees Angle-of-Attack

Figure 7.3 depicts the variation of $\left(C_f \ Re_L\right)_\omega$ in the windward plane for the axisymmetric analogue using Hall's method and Wang's fully three-dimensional approach for an angle-of-attack of 12° (Ref. 5). The two methods compare reasonably well.

7.3. Sphere-Ogive-Cylinder

This configuration, whose geometry is depicted in Figure 5.1, was input to the program in dimensionless form. The normalizing quantity was the cylinder radius which measured 3.8 inches. The total non-dimensional body length was 17.5. This geometry was investigated at 45° angle-of-attack. The experimental pressure data consisted of discrete pressure coefficients distributed along 30 axial stations, each having 10 circumferential stations. Pressure data for the region $0 \leq x \leq 13.75$ was calculated by the USSAERO panel method while the pressure over the section $0.92 \leq x \leq 17.5$ consisted of actual experimental pressures obtained from the wind tunnel. Before implementing these pressures into the computational code, it was necessary to smooth and interpolate the data in the region where the pressures overlapped. Interpolated data were used to form additional axial stations near the nose since large pressure gradients are present on the forward portion of the body. This required the addition of four more axial stations in that region (the resulting pressure coefficients as well as the remaining program inputs are presented in Appendix A.22).

The technique developed to represent the pressure distribution on the spherical cap by a 10-term Fourier cosine series was found to perform only marginally. The series provided continuity in the pressure coefficient across the interface (when the quadratic spline was first employed) but did not necessarily provide continuity in the related derivatives. In instances

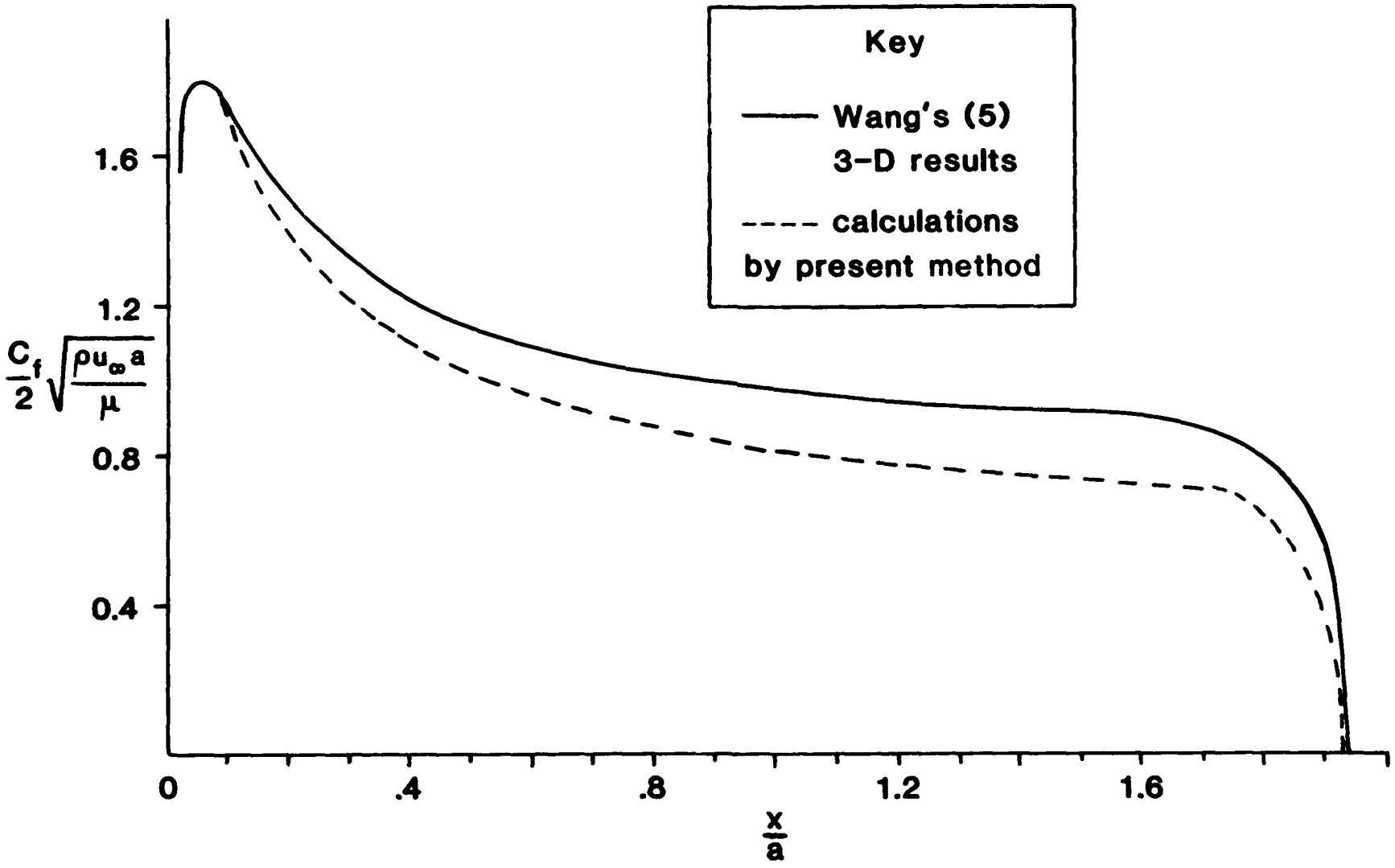


Figure 7.3. Skin Friction Distribution in Windward Plane of Ellipsoid of Revolution with Thickness Ratio 1/4 at 12 Degrees Angle-of-Attack

in which the pressure derivatives were not continuous across the interface, the one-dimensional quadratic spline technique was extended from the interface to the stagnation point to model the pressure variation on the spherical cap (see SURFACE PRESSURE DISTRIBUTION). The resulting pressure variation was a function of the circumferential position on the interface. Despite this, the streamlines in this region were still assumed to follow spherical meridians.

The doubly quadratic spline technique employed in this computer program was found to model the pressure coefficient variation quite satisfactorily. This technique requires that the axial derivative of the pressure coefficient at the interface and body end for each circumferential plane be known. While the pressure coefficient across the interface was continuous, in most cases the axial derivative was not and, hence, was also supplied as program input (rather than calculated in the program).

The streamline angle and its circumferential derivative were calculated in this case by numerically integrating Equations (2.18) and (2.39). These equations are functions of the inviscid edge velocity, the pressure coefficient and its derivatives. These parameters were provided by the spline fit. Although the calculated second derivatives of the pressure coefficient in the circumferential direction are constant in the interval in which the quadratic is used, they were found to be accurate enough to be used in the integration of Equation (2.39). Table 7.5 provides a comparison between the calculated separation points using Hall's and Blottner's methods. The corresponding separation points agree very well. Note that for the cases of $\beta = 145^{\circ}$ and

Table 7.5. Comparison of Separation Points Between Hall's and Blottner's Methods on Sphere-Ogive-Cylinder Configuration at 45° Angle-of-Attack

Method	Beta	Points Added	Computer Time (sec)	Separation X/R_c	Separation ϕ	Separation ψ
Hall	50	39	9	0.2158	119.15	--
	75	0	8	0.1777	125.32	--
	100	3	8	0.1562	121.34	--
	145	5	8	--	--	115.14
	160	27	8	--	--	119.64
Blottner	50	3	8	0.2159	119.21	--
	75	3	8	0.1763	123.75	--
	100	1	8	0.1564	121.66	--
	145	2	8	--	--	115.16
	160	0	8	--	--	119.25

160°, the flow separates while on the spherical cap. If the pressure distribution had been truly axisymmetric, the angle of separation, ψ, would have been identical in each case. The computational time required for this configuration was greater than that for the ellipsoid of revolution though still quite reasonable.

Information relative to the step sizes and spacings is given in Table 7.6. The step size on the spherical cap was $\Delta\psi = 2°$ in each case.

Table 7.6. Computational Step Sizes and Spacings for Results in Table 7.5

Method	Beta	Δs	Δn or $\Delta \eta$	Steps to Separation
Hall	50	0.001	0.003846	109
	75	0.004	0.015385	50
	100	0.004	0.007692	52
	145	---	0.007692	
	160	---	0.003846	55
Blottner	50	0.001	0.1153846	136
	75			
	100			
	145	---	0.1153846	59
	160	---	0.1153846	60

SECTION 8

CONCLUDING REMARKS

A method is developed for calculating laminar boundary layers along inviscid surface streamlines on axisymmetric bodies at angles of attack in incompressible flow. By application of the axisymmetric analogue concept in the present technique, a substantial savings in computer time over fully three-dimensional boundary layer techniques may be realized.

The boundary layer integration techniques of Hall and Blottner were found to compare exceptionally well with each other on each of the geometries investigated. Results generated on the windward plane of an ellipsoid of revolution with thickness ratio 1/4 and angle of attack of 12° compared satisfactorily with results generated by a fully three-dimensional technique. The separation points calculated by the present technique for a variety of streamlines on the same ellipsoid of revolution at 30° angle of attack were in fair agreement to those generated by a three-dimensional technique. The comparison was generally better on the windside of the body.

The series expression used to model the pressure coefficient on the spherical cap of the sphere-ogive-cylinder configuration performed only satisfactory. The technique preserved continuity in the pressure coefficient across the interface (the point at which the quadratic spline technique was implemented) but did not provide continuity of the axial derivative. To circumvent this problem, the one-dimensional quadratic spline was extended to the stagnation point on the spherical cap. Despite this, the assumption of spherical streamlines was still made with reasonable accuracy. The doubly quadratic spline representation of the

pressure coefficient on the afterbody was found to perform quite well as long as an adequate number of pressure stations were input. Oscillations in the pressure function were generally less frequent than might be expected if other techniques had been used.

The relative inexpense, coupled with reasonable accuracy makes the present method attractive for preliminary design studies. Further comparisons with fully three-dimensional boundary layer calculations are necessary in order to more thoroughly evaluate the applicability of the axisymmetric analogue in subsonic flow.

LIST OF REFERENCES

1. Smith, A. M. O. and Clutter, D. W., "Solutions of the Incompressible Laminar Boundary-Layer Equations," AIAA Journal, Vol. 1, No. 9, September 1963.

2. Blottner, F. G., "Introduction to Computational Techniques for Boundary Layers," SAND 79-0893, Sandia Laboratories Publication, Albuquerque, New Mexico, September 1979.

3. Blottner, F. G. and Ellis, M. A., "Finite-Difference Solution of the Incompressible Three-Dimensional Boundary-Layer Equations for a Blunt Body," Computers and Fluids, Vol. 1, 1973.

4. Hall, M. G., "A Numerical Method for Calculating Steady Three-Dimensional Laminar Boundary Layers," RAE TR-67145, June 1967.

5. Wang, K. C., "Three-Dimensional Boundary Layers Near the Plane of Symmetry of a Spheroid at Incidence," Journal of Fluid Mechanics, Vol. 43, Part 1, October 1969.

6. Cooke, J. C. and Hall, M. G., "Boundary-Layers in Three-Dimensions," Progress in Aeronautical Sciences, Vol. 2, 1962, Pergamon Press, Inc., Elmsford, NY.

7. Vollmers, H., "Integration of Streamlines from Measured Static Pressure Fields on a Surface," AIAA Journal, Vol. 20, No. 10, October 1982.

8. DeJarnette, F. R. and Hamilton, H. H., "Inviscid Surface Streamlines and Heat Transfer on Shuttle-Type Configurations," Journal of Spacecraft and Rockets, Vol. 10, No. 6, May 1973.

9. DeJarnette, F. R., "Calculation of Inviscid Surface Streamlines and Heat Transfer on Shuttle-Type Configurations," NASA CR-111921, August 1971.

10. Geissler, W., "Three-Dimensional Laminar Bounary Layer Over a Body of Revolution at Incidence and With Separation," AIAA Journal, Vol. 12, No. 12, December 1974.

11. Howarth, L., "The Boundary Layer in Three-Dimensional Flow, Pt. 2: The Flow Near a Stagnation Point," Philosophical Magazine, Vol. 42, Ser. 7, No. 335, December 1951.

12. Gear, C. W., Numerical Initial Value Problems in Ordinary Differential Equations. Prentice-Hall, Inc., Englewood Cliffs, NJ, 1971.

13. Wang, K. C., "Boundary Layer Over a Blunt Body at High Incidence With an Open-Type of Separation," Martin Marietta Laboratories, Baltimore, Maryland, October 1973.

14. DeJarnette, F. R. and Davis, R. M., "A Simplified Method for Calculating Laminar Heat Transfer Over Bodies at an Angle of Attack," NASA TND-4720, August 1968.

LIST OF SYMBOLS

a_n, b_n, c_n, d_n	coefficients defined in Equation (4.3)
A_1, B_1, C_1, D_1, E_1 A_2, B_2, C_2, D_2, E_2	coefficients for finite-difference boundary-layer equations, defined in Equations (3.8), (3.9), (3.12), (3.13)
$\hat{e}_{11}$	unit vectors on body surface along body meridian given by Equation (2.2)
$\hat{e}_s, \hat{e}_\beta, \hat{e}_n$	unit vectors in streamline coordinate system given by Equations (2.3), (2.4) and (2.1)
$\hat{e}_x, \hat{e}_r, \hat{e}_\phi$	unit vectors in cylindrical coordinate system, (see Figure 2.1)
C_f	skin friction coefficient, $\frac{2\tau_\omega}{\rho u_\infty^2}$
$\frac{D}{Ds}$	total derivative along streamline
F	ratio of local velocity to velocity at boundary-layer edge, as defined by Equation (3.6), dimensionless
h	scale factor in β direction, dimensionless
K	arbitrary constant for Equation (3.1) and (3.2)
L	body length, dimensionless
n	coordinate normal to body surface and streamline
p	dimensional pressure, lb/ft^2 or N/m^2
r	body radius, dimensionless
R_c	radius of cylinder in sphere-ogive-cylinder configuration, dimensionless
R_{per}	radius of spherical cap, dimensionless
R_{e_L}	freestream Reynold's number, $\frac{\rho u_\infty L}{\mu}$
s	distance along streamline, dimensionless
u	local fluid velocity in boundary layer (in direction of a streamline) ft/sec or m/sec

U	inviscid fluid velocity, ft/sec or m/sec
v	local fluid velocity normal to streamline and body surface, ft/sec or m/sec
V	parameter defined by Equation (3.5)
x,y,z	body geometry coordinate axes (see Figure 2.1)
Y(1)	axial position, x, dimensionless, ft or m
Y(2)	circumferential angle, ϕ, rads
Y(3)	streamline angle, θ, rads
Y(4)	$\left.\frac{\partial\theta}{\partial\phi}\right\|_x$
Y(5)	$\ell n \left.\frac{\partial\phi}{\partial\beta}\right\|_x$
Y(6)	transformed streamline coordinate, ξ
α	angle of attack, degrees
β	coordinate normal to streamline and tangent to body
$\bar{\beta}$	pressure gradient parameter, $\frac{2\xi}{u_e}\frac{du_e}{d\xi}$
Γ	body angle, radians (see Figure 2.2)
ξ	transformed streamline coordinate as defined in (3.1)
η	transformed coordinate normal to body surface defined in (3.2)
ϕ	circumferential angle (see Figure 2.1), rads
θ	streamline angle (see Figure 2.3), rads
ψ	angle between stagnation line and radius vector (see Figure 4.1), rads
ρ	density, slug/ft^3 or kg/m^3
μ	coefficient of viscosity, slug/ft-sec or kg/m-s

Subscripts

∞	freestream conditions
e	edge of the boundary layer
eff	effective value
ω	at the wall
o	initial value
i	streamline grid index
I	value at sphere-afterbody interface
j	normal grid index
SP	value at stagnation point

Superscripts

$\bar{\ }$	*denotes a quantity from a previous iteration*
*	denotes a dimensionless quantity

APPENDIX A. EQUATIONS

1. Derivation of Equations (3.3) and (3.4)

Blottner's method involves the boundary layer equations written in F-V similarity form. These are obtained by application of the Levy-Lees transformation which is defined as

$$\xi(s) = K\rho\mu u_\infty \int_0^s \frac{u_e}{u_\infty} r^2 \, ds \qquad (A.1.1)$$

and

$$\eta(s,n) = \frac{u_e \rho r n}{\sqrt{2\xi}} \sqrt{K} \qquad (A.1.2)$$

for incompressible flow. The transformation operators may be constructed and expressed as

$$\frac{\partial}{\partial s} = \frac{u_e}{u_\infty} r^2 \frac{\partial}{\partial \xi} + \frac{\partial \eta}{\partial s} \frac{\partial}{\partial \eta} \qquad (A.1.3)$$

and

$$\frac{\partial}{\partial n} = \frac{u_e \rho r}{\sqrt{2\xi}} \frac{1}{\sqrt{\rho\mu u_\infty}} \frac{\partial}{\partial \eta} \qquad (A.1.4)$$

where the arbitrary constant, K, has been assigned the value

$$K = 1/\rho\mu u_\infty \quad . \qquad (A.1.5)$$

The dependent variable in Blottner's equations is defined as

$$F = u/u_e \quad . \tag{A.1.6}$$

Application of each transformation operator and the definition of F to the continuity equation,

$$\frac{\partial(ru)}{\partial s} + r\frac{\partial v}{\partial n} = 0 \tag{A.1.7}$$

yields

$$\frac{u_e}{u_\infty} r^2 \frac{\partial(ru_eF)}{\partial \xi} + \frac{\partial \eta}{\partial s} ru_e \frac{\partial F}{\partial \eta} + \frac{u_e r^2 \rho}{\sqrt{2\xi}} \frac{1}{\sqrt{\rho\mu u_\infty}} \frac{\partial v}{\partial \eta} = 0 \quad . \tag{A.1.8}$$

Note that neither u_e nor r are functions of the normal coordinate n. Expanding this equation yields

$$\frac{u_e}{u_\infty} r^2 \left\{ ru_e \frac{\partial F}{\partial \xi} + u_e F \frac{dr}{d\xi} + Fr \frac{du_e}{d\xi} \right\} + ru_e \frac{\partial \eta}{\partial s} \frac{\partial F}{\partial \eta} + \frac{u_e r^2 \rho}{\sqrt{2\xi}} \frac{1}{\sqrt{\rho\mu u_\infty}} \frac{\partial v}{\partial \eta} = 0 \quad . \tag{A.1.9}$$

This equation may be rewritten as

$$\frac{u_e^2 r^3}{u_\infty} \left\{ \frac{\partial F}{\partial \xi} + \frac{F}{r}\frac{dr}{d\xi} + \frac{F}{u_e}\frac{du_e}{d\xi} \right\} + ru_e \frac{\partial \eta}{\partial s} \frac{\partial F}{\partial \eta} + \frac{u_e r^2 \rho}{\sqrt{2\xi}} \frac{1}{\sqrt{\rho\mu u_\infty}} \frac{\partial v}{\partial \eta} = 0 \quad . \tag{A.1.10}$$

By application of the product rule, this equation may be written as

$$\frac{\partial F}{\partial \xi} + \frac{F}{r}\frac{dr}{d\xi} + \frac{F}{u_e}\frac{du_e}{d\xi} + \frac{u_\infty}{u_e^2 r^3} \frac{\partial}{\partial \eta} \left\{ ru_e \frac{\partial \eta}{\partial s} F + \frac{u_e r^2 \rho v}{\sqrt{2\xi}\sqrt{\rho\mu u_\infty}} \right\}$$

$$- \frac{u_\infty}{u_e r^2} F \frac{\partial}{\partial \eta} \left(\frac{\partial \eta}{\partial s} \right) = 0 \quad . \tag{A.1.11}$$

The derivative in the last term of this equation may be rewritten as:

$$\frac{\partial}{\partial \eta}\left(\frac{\partial \eta}{\partial s}\right) = \frac{\partial}{\partial n}\frac{\partial n}{\partial \eta}\left(\frac{\partial \eta}{\partial s}\right) = \frac{\sqrt{2\xi}\sqrt{\rho\mu u_\infty}}{u_e r \rho}\left\{\frac{du_e}{ds}\frac{r\rho}{\sqrt{2\xi}\sqrt{\rho\mu u_\infty}}\right.$$

$$\left. + \frac{dr}{ds}\frac{u_e\rho}{\sqrt{2\xi}\sqrt{\rho\mu u_\infty}} - \frac{1}{2\xi}\frac{u_e r\rho}{\sqrt{2\xi}\sqrt{\rho\mu u_\infty}}\frac{d\xi}{ds}\right\} \quad , \qquad (A.1.12)$$

which becomes, upon making use of Equations (A.1.1) and (A.1.5),

$$\frac{\partial}{\partial \eta}\left(\frac{\partial \eta}{\partial s}\right) = \frac{r^2}{u_\infty}\frac{du_e}{d\xi} + \frac{u_e r}{u_\infty}\frac{dr}{d\xi} - \frac{u_e r^2}{2u_\infty \xi} \quad .$$

Substitution of this relation into Equation (A.1.11) yields

$$\frac{\partial F}{\partial \xi} + \frac{F}{r}\frac{dr}{d\xi} + \frac{F}{u_e}\frac{du_e}{d\xi} + \frac{\partial}{\partial \eta}\left\{\frac{u_\infty}{u_e r^2}\frac{\partial \eta}{\partial s}F + \frac{u_\infty \rho v}{u_e r}\frac{1}{\sqrt{2\xi}\sqrt{\rho\mu u_\infty}}\right\}$$

$$- \frac{F}{u_e}\frac{du_e}{d\xi} - \frac{F}{r}\frac{dr}{d\xi} + \frac{F}{2\xi} = 0 \quad . \qquad (A.1.13)$$

This equation may then be written as

$$2\xi\frac{\partial F}{\partial \xi} + F + \frac{\partial V}{\partial \eta} = 0 \quad , \qquad (A.1.14)$$

where

$$V = 2\xi\left(F\frac{\partial \eta}{\partial s}\frac{u_\infty}{u_e r^2} + \frac{\rho v u_\infty}{\sqrt{2\xi}\sqrt{\rho\mu u_\infty}}\frac{1}{u_e r}\right) \quad . \qquad (A.1.15)$$

Application of the transformation operators in (A.1.3) and (A.1.4) and the definition of F to the momentum equation

$$u \frac{\partial u}{\partial s} + v \frac{\partial u}{\partial n} = u_e \frac{du_e}{ds} + \frac{\mu}{\rho} \frac{\partial^2 u}{\partial n^2} \qquad (A.1.16)$$

yields

$$u_e F \left\{ \frac{u_e r^2}{u_\infty} \frac{\partial (u_e F)}{\partial \xi} + \frac{\partial \eta}{\partial s} u_e \frac{\partial F}{\partial \eta} \right\} + \frac{v u_e^2 r \rho}{\sqrt{2\xi} \sqrt{\rho \mu u_\infty}} \frac{\partial F}{\partial \eta} = \frac{u_e^2 r^2}{u_\infty} \frac{du_e}{d\xi}$$

$$+ \frac{u_e^3 r^2 \rho^2}{2\xi \rho \mu u_\infty} \frac{\mu}{\rho} \frac{\partial^2 F}{\partial \eta^2} \quad . \qquad (A.1.17)$$

This may be expanded to give

$$\left\{ \frac{(u_e r F)^2}{u_\infty} \frac{du_e}{d\xi} + \frac{u_e^3 r^2 F}{u_\infty} \frac{\partial F}{\partial \xi} + u_e^2 F \frac{\partial \eta}{\partial s} \frac{\partial F}{\partial \eta} \right\} + \frac{v u_e^2 r \rho}{\sqrt{2\xi} \sqrt{\rho \mu u_\infty}} \frac{\partial F}{\partial \eta} = \frac{(u_e r)^2}{u_\infty} \frac{du_e}{d\xi}$$

$$+ \frac{u_e^3 r^2}{2\xi u_\infty} \frac{\partial^2 F}{\partial \eta^2} \quad . \qquad (A.1.18)$$

This equation may then be rearranged to yield

$$\frac{2\xi}{u_e} F^2 \frac{du_e}{d\xi} + 2\xi F \frac{\partial F}{\partial \xi} + \frac{2\xi F u_\infty}{u_e r^2} \frac{\partial \eta}{\partial s} \frac{\partial F}{\partial \eta} + \frac{v}{\sqrt{2\xi}} \frac{2\xi u_\infty}{u_e r} \frac{\rho}{\sqrt{\rho \mu u_\infty}} \frac{\partial F}{\partial \eta} = \frac{2\xi}{u_e} \frac{du_e}{d\xi} + \frac{\partial^2 F}{\partial \eta^2} \quad .$$

$$(A.1.19)$$

or, finally,

$$2\xi F \frac{\partial F}{\partial \xi} + (F^2 - 1) \beta + V \frac{\partial F}{\partial \eta} - \frac{\partial^2 F}{\partial \eta^2} = 0 \qquad (A.1.20)$$

where β is the pressure gradient parameter and is defined as

$$\beta = \frac{2\xi}{u_e} \frac{du_e}{d\xi} \quad .$$

2. Derivation of Equations (3.8) and (3.9)

At all points off the stagnation point, the full system of F-V equations must be solved. The continuity equation,

$$2\xi \frac{\partial F}{\partial \xi} + \frac{\partial V}{\partial \eta} + F = 0 \qquad (A.2.1)$$

is evaluated at the point (i + 1/2, j - 1/2) and may be expressed as

$$2\xi_{i+\frac{1}{2}} \frac{\left.\frac{\partial F}{\partial \xi}\right|_{i+\frac{1}{2},j} + \left.\frac{\partial F}{\partial \xi}\right|_{i+\frac{1}{2},j-1}}{2} + \left.\frac{\partial V}{\partial \eta}\right|_{i+\frac{1}{2},j-\frac{1}{2}}$$

$$+ \frac{(F_{i+1,j} + F_{i+1,j-1} + F_{i,j} + F_{i,j-1})}{4} = 0$$

Substituting second-order accurate expressions for the appropriate quantities yields

$$\frac{\xi_{i+\frac{1}{2}}}{\Delta\xi}\left(F_{i+1,j} - F_{i,j} + F_{i+1,j-1} - F_{i,j-1}\right) + \frac{(V_{i+\frac{1}{2},j} - V_{i+\frac{1}{2},j-1})}{\Delta\eta}$$

$$+ \frac{(F_{i+1,j} + F_{i+1,j-1} + F_{i,j} + F_{i,j-1})}{4} = 0 \quad .$$

After rearranging, the continuity equation may be expressed as

$$A_2 F_{i+1,j-1} + B_2 F_{i+1,j} + C_2 V_{i+\frac{1}{2},j-1} + E_2 V_{i+\frac{1}{2},j} = D_2 \qquad (A.2.2)$$

where

$$A_2 = \Delta\eta(\tfrac{1}{4} + \xi_{i+\frac{1}{2}}/\Delta\xi)$$

$$B_2 = \Delta\eta(\tfrac{1}{4} + \xi_{i+\frac{1}{2}}/\Delta\xi)$$

$$C_2 = -1$$

$$E_2 = 1$$

and

$$D_2 = \Delta\eta(-\tfrac{1}{4} + \xi_{i+\frac{1}{2}}/\Delta\xi)(F_{i,j} + F_{i,j-1}) \quad .$$

The momentum equation,

$$2\xi F \frac{\partial F}{\partial \xi} + V \frac{\partial F}{\partial \eta} + \bar{\beta}\,(F^2 - 1) - \frac{\partial^2 F}{\partial \eta^2} = 0 \qquad \text{(A.2.3)}$$

is evaluated at $(i+\frac{1}{2},j)$. The terms of Equation (A.2.3) become, respectively,

$$2\xi F \left.\frac{\partial F}{\partial \xi}\right|_{i+\frac{1}{2},j} = 2\xi_{i+\frac{1}{2}} \left[\frac{F_{i+1,j} + F_{i,j}}{2}\right]\left[\frac{F_{i+1,j} - F_{i,j}}{\Delta\xi}\right] \quad ,$$

$$V \left.\frac{\partial F}{\partial \eta}\right|_{i+\frac{1}{2},j} = \frac{V_{i+\frac{1}{2},j}}{2}\left[\left.\frac{\partial F}{\partial \eta}\right|_{i+1,j} + \left.\frac{\partial F}{\partial \eta}\right|_{i,j}\right] \quad ,$$

$$\bar{\beta}(F^2 - 1)_{i+\frac{1}{2},j} = \bar{\beta}\left[\frac{F^2_{i+1,j} + F^2_{i,j}}{2} - 1\right] \quad ,$$

$$\left.\frac{\partial^2 F}{\partial \eta^2}\right|_{i+\frac{1}{2},j} = \frac{1}{2}\left[\left.\frac{\partial^2 F}{\partial \eta^2}\right|_{i+1,j} + \left.\frac{\partial^2 F}{\partial \eta^2}\right|_{i,j}\right] \quad . \qquad \text{(A.2.4)}$$

Linearizing the first three terms using the Newton-Raphson method yields

$$2\xi_{i+\frac{1}{2}} F_{i+\frac{1}{2},j} \frac{\partial F}{\partial \xi}_{i+\frac{1}{2},j} = \frac{\xi_{i+\frac{1}{2}}}{\Delta\xi} \left(2\bar{F}_{i+1,j} F_{i+1,j} - \bar{F}^2_{i+1,j} - F^2_{i,j}\right) ,$$

$$V \left.\frac{\partial F}{\partial \eta}\right|_{i+\frac{1}{2},j} = \frac{1}{2} V_{i+\frac{1}{2},j} \left[\left.\frac{\partial \bar{F}}{\partial \eta}\right|_{i+1,j} + \left.\frac{\partial F}{\partial \eta}\right|_{i,j}\right] + \frac{1}{2} \bar{V}_{i+\frac{1}{2},j} \left.\left(\frac{\partial F}{\partial \eta}\right)\right|_{i+1,j}$$

$$- \frac{1}{2} \bar{V}_{i+\frac{1}{2},j} \left.\frac{\partial \bar{F}}{\partial \eta}\right|_{i+1,j} ,$$

$$\bar{\beta}(F^2 - 1)\big|_{i+\frac{1}{2},j} = \frac{\bar{\beta}_{i+\frac{1}{2}}}{2} \left(2F_{i+1,j} \bar{F}_{i+1,j} - \bar{F}^2_{i+1,j} + F^2_{i,j} - 1\right) . \qquad (A.2.5)$$

Substituting second-order accurate finite difference expressions in the second of Equations (A.2.5) and the last of Equations (A.2.4) gives, respectively,

$$V \left.\frac{\partial F}{\partial \eta}\right|_{i+\frac{1}{2},j} = \frac{1}{2} V_{i+\frac{1}{2},j} \left[\frac{\bar{F}_{i+1,j+1} - \bar{F}_{i+1,j-1}}{\Delta\eta} + \frac{F_{i,j+1} - F_{i,j-1}}{\Delta\eta}\right]$$

$$+ \frac{1}{2} \bar{V}_{i+\frac{1}{2},j} \left[\frac{F_{i+1,j+1} - F_{i+1,j-1}}{2\Delta\eta}\right] - \frac{1}{2} \bar{V}_{i+\frac{1}{2},j} \left[\frac{\bar{F}_{i+1,j+1} - \bar{F}_{i+1,j-1}}{2\Delta\eta}\right]$$

and

$$\left.\frac{\partial^2 F}{\partial \eta^2}\right|_{i+\frac{1}{2},j} = \frac{1}{2} \left[\frac{F_{i+1,j+1} - 2F_{i+1,j} + F_{i+1,j-1}}{\Delta\eta^2} + \frac{F_{i,j+1} - 2F_{i,j} + F_{i,j-1}}{\Delta\eta^2}\right] .$$

After substituting the appropriate expressions into Equation (A.2.3), solving for the unbarred quantities and rearranging, the equation may be written as

$$A_1 F_{i+1,j-1} + B_1 F_{i+1,j} + C_1 F_{i+1,j+1} + E_1 V_{i+\frac{1}{2},j} = D_1$$

where

$$A_1 = -\frac{1}{2}\left(1 + \frac{1}{2}\,\Delta\eta\,\bar{V}_{i+\frac{1}{2},j}\right)$$

$$B_1 = 1 + \Delta\eta^2\,\bar{F}_{i+1,j}\left(\bar{\beta}_{i+\frac{1}{2}} + 2\xi_{i+\frac{1}{2}}/\Delta\xi\right)$$

$$C_1 = -\frac{1}{2}\left(1 - \frac{1}{2}\,\Delta\eta\,\bar{V}_{i+\frac{1}{2},j}\right)$$

$$E_1 = \frac{1}{4}\,\Delta\eta\left(F_{i,j+1} - F_{i,j-1} + \bar{F}_{i+1,j+1} - \bar{F}_{i+1,j-1}\right)$$

and

$$D_1 = \frac{1}{2}\left(F_{i,j+1} - 2F_{i,j} + F_{i,j-1}\right) + \frac{1}{2}\,\Delta\eta^2\,\bar{\beta}_{i+\frac{1}{2}}\left[\left(1 + \bar{F}^2_{i+1,j}\right) + \left(1 - F^2_{i,j}\right)\right] + \frac{1}{4}\,\Delta\eta\,\bar{V}_{i+\frac{1}{2},j}\left(\bar{F}_{i+1,j+1} - \bar{F}_{i+1,j-1}\right) + \Delta\eta^2\,\xi_{i+\frac{1}{2}}\left(\bar{F}^2_{i+1,j} + \bar{F}^2_{i,j}\right)/\Delta\xi \quad .$$

3. Derivation of Finite-Difference F-V Similarity Equations at Stagnation Point

At the stagnation point $\xi = 0$ and the F-V equations reduce to

$$\frac{\partial V}{\partial \eta} + F = 0 \qquad (A.3.1)$$

for the continuity equation and

$$V \frac{\partial F}{\partial \eta} + \bar{\beta} (F^2 - 1) - \frac{\partial^2 F}{\partial \eta^2} = 0 \qquad (A.3.2)$$

for the momentum equation.

The continuity equation is evaluated at $(j - 1/2)$ and may be expressed as

$$\frac{V_j - V_{j-1}}{\Delta \eta} + \frac{(F_j + F_{j-1})}{2} = 0 \quad .$$

This may be rewritten as

$$V_j = V_{j-1} - \frac{\Delta \eta}{2} (F_j + F_{j-1}) \quad . \qquad (A.3.3)$$

The momentum equation is evaluated at (j) after first being linearized using the Newton-Ranhson technique. The equation may then be written as

$$V \frac{\overline{\partial F}}{\partial \eta} + \bar{V} \frac{\partial F}{\partial \eta} - \bar{V} \frac{\overline{\partial F}}{\partial \eta} + \bar{\beta} (2F\bar{F} - \bar{F}^2 - 1) - \frac{\partial^2 F}{\partial \eta^2} = 0 \quad ,$$

where the barred quantities denote the expressions from the previous iteration. Substituting appropriate finite difference approximations for each of the above terms yields

$$V_j \left[\frac{\bar{F}_{j+1} - \bar{F}_{j-1}}{2\Delta\eta}\right] + \bar{V}_j \left[\frac{F_{j+1} - F_{j-1}}{2\Delta\eta}\right] - \bar{V}_j \left[\frac{\bar{F}_{j+1} - \bar{F}_{j-1}}{2\Delta\eta}\right]$$

$$+ \bar{\beta}\,(2F\bar{F}_j - \bar{F}_j^2 - 1) - \left[\frac{F_{j+1} - 2F_j + F_{j-1}}{\Delta\eta^2}\right] = 0 \quad .$$

Solution of this equation for the unbarred quantities yields

$$A_1 F_{j-1} + B_1 F_j + C_1 F_{j+1} + E_1 V_j = D_1 \quad ,$$

where

$$A_1 = 2 + \Delta\eta\, \bar{V}_j$$

$$B_1 = -(4 + 4\,\bar{\beta}\Delta\eta^2\, \bar{F}_j)$$

$$C_1 = 2 - \Delta\eta\, \bar{V}_j$$

$$E_1 = \Delta\eta(\bar{F}_{j-1} - \bar{F}_{j+1})$$

and

$$D_1 = -\Delta\eta\, \bar{V}_j\, (\bar{F}_{j+1} - \bar{F}_{j-1}) - \bar{\beta}(1 + \bar{F}_j^2)\, 2\Delta\eta^2 \quad .$$

4. Derivation of Equations (3.10) and (3.11)

Hall's method involves solving the boundary layer equations written in terms of dimensionless primitive variables. The transformations used to obtain these equations are

$$s^* = s/L$$

$$n^* = \sqrt{Re_L}\, n/L$$

$$u^* = u/u_\infty$$

$$v^* = \sqrt{Re_L}\, v/u_\infty$$

$$r^* = r/L$$

and

$$\sqrt{Re_L} = \sqrt{\rho \frac{u_\infty L}{\mu}}$$

and a star denotes a dimensionless quantity. Accordingly, the transformation operators are constructed as follows

$$\frac{\partial}{\partial s} = \frac{1}{L} \frac{\partial}{\partial s^*}$$

and

$$\frac{\partial}{\partial n} = \frac{\sqrt{Re_L}}{L} \frac{\partial}{\partial n^*} \quad .$$

Application of each transformation operator to the continuity equation,

$$\frac{\partial(ru)}{\partial s} + r\,\frac{\partial v}{\partial n} = 0$$

yields

$$\frac{u_\infty L}{L}\,\frac{\partial(r^*u^*)}{\partial s^*} + r^*L\,\frac{\sqrt{R_{e_L}}}{L}\,\frac{\partial}{\partial n^*}\left(\frac{u_\infty v^*}{\sqrt{R_{e_L}}}\right) = 0$$

which will simplify to

$$\frac{\partial(r^*u^*)}{\partial s^*} + r^*\,\frac{\partial v^*}{\partial n^*} = 0 \quad .$$

Note that r is not a function of the normal coordinate n.

Application of each of the transformation operators to the momentum equation,

$$u\,\frac{\partial u}{\partial s} + v\,\frac{\partial u}{\partial n} = u_e\,\frac{du_e}{ds} + \frac{\mu}{\rho}\,\frac{\partial^2 u}{\partial n^2}$$

yields

$$\frac{u_\infty^2}{L}\,u^*\,\frac{\partial u^*}{\partial s^*} + \frac{v^*u_\infty^2}{\sqrt{R_{e_L}}}\,\frac{\sqrt{R_{e_L}}}{L}\,\frac{\partial u^*}{\partial n^*} = \frac{u_\infty^2}{L}\,u_e^*\,\frac{du_e^*}{ds^*} + \frac{\mu}{\rho}\,u_\infty\,\frac{R_{e_L}}{L}\,\frac{\partial^2 u^*}{\partial n^{*2}}$$

which may be simplified to

$$u^*\,\frac{\partial u^*}{\partial s^*} + v^*\,\frac{\partial u^*}{\partial n^*} = u_e^*\,\frac{du_e^*}{ds^*} + \frac{\partial^2 u^*}{\partial n^{*2}} \quad .$$

5. Derivation of Equations (3.12) and (3.13)

The nondimensional continuity equation,

$$\frac{\partial(hu)}{\partial s} + h\,\frac{\partial v}{\partial n} = 0$$

(the stars have been deleted for clarity) is evaluated at the point (i + 1/2, j - 1/2) in the computational grid. The equation may then be written as

$$\frac{1}{2}\left[\left.\frac{\partial(hu)}{\partial s}\right|_{i+\frac{1}{2},j} + \left.\frac{\partial(hu)}{\partial s}\right|_{i+\frac{1}{2},j-1}\right] + h_{i+\frac{1}{2}}\left.\frac{\partial v}{\partial n}\right|_{i+\frac{1}{2},j-\frac{1}{2}} = 0$$

substituting second-order accurate expressions for the appropriate terms yields

$$\frac{1}{2}\left[\frac{(hu)_{i+1,j} - (hu)_{i,j}}{\Delta s} + \frac{(hu)_{i+1,j-1} - (hu)_{i,j-1}}{\Delta s}\right] + h_{i+\frac{1}{2}}\left[\frac{v_{i+\frac{1}{2},j} - v_{i+\frac{1}{2},j-1}}{\Delta n}\right] = 0 \quad .$$

Solving for $v_{i+\frac{1}{2},j}$ gives

$$v_{i+\frac{1}{2},j} = v_{i+\frac{1}{2},j-1} - \frac{\Delta n}{2\Delta s h_{i+\frac{1}{2}}}\left[h_{i+1}\,(u_{i+1,j} + u_{i+1,j-1}) - h_i\,(u_{i,j} + u_{i,j-1})\right].$$

After rearranging, the continuity equation may be rewritten as

$$A_2 u_{i+1,j} + B_2 u_{i+1,j-1} + C_2 v_{i+\frac{1}{2},j} + E_2 v_{i+\frac{1}{2},j-1} = D_2$$

where

$$A_2 = \frac{\Delta nh_{i+1}}{2\Delta sh_{i+1/2}}$$

$$B_2 = \frac{\Delta nh_{i+1}}{2\Delta sh_{i+1/2}}$$

$$C_2 = 1$$

$$E_2 = -1$$

and

$$D_2 = \frac{\Delta nh_i}{2\Delta sh_{i+1/2}} \left(u_{i,j} + u_{i,j-1}\right) \quad .$$

The nondimensional momentum equation,

$$u \frac{\partial u}{\partial s} + v \frac{\partial u}{\partial n} = u_e \frac{du_e}{ds} + \frac{\partial^2 u}{\partial n^2}$$

is evaluated at the point (i + 1/2,j). Taken term by term, this may be written as

$$u_{i+1/2,j} = \frac{1}{2} \left(u_{i+1,j} + u_{i,j}\right)$$

$$\left.\frac{\partial u}{\partial s}\right|_{i+1/2,j} = \frac{u_{i+1,j} - u_{i,j}}{\Delta s}$$

$$\left.\frac{\partial u}{\partial n}\right|_{i+1/2,j} = \frac{1}{2} \left[\left.\frac{\partial u}{\partial n}\right|_{i+1,j} + \left.\frac{\partial u}{\partial n}\right|_{i,j}\right]$$

$$\left.\frac{\partial^2 u}{\partial n^2}\right|_{i+\frac{1}{2},j} = \frac{1}{2}\left[\left.\frac{\partial^2 u}{\partial n^2}\right|_{i+1,j} + \left.\frac{\partial^2 u}{\partial n^2}\right|_{i,j}\right] .$$

Substituting second-order accurate expressions for the remaining derivatives yields

$$\left.\frac{\partial u}{\partial n}\right|_{i+\frac{1}{2},j} = \frac{1}{4\Delta n}\left[u_{i+1,j+1} - u_{i+1,j-1} + u_{i,j+1} - u_{i,j-1}\right]$$

and

$$\left.\frac{\partial^2 u}{\partial n^2}\right|_{i+\frac{1}{2},j} = \frac{1}{2\Delta n^2}\left[u_{i+1,j+1} - 2u_{i+1,j} + u_{i+1,j-1} + u_{i,j+1} - 2u_{i,j} + u_{i,j-1}\right] .$$

Linearizing using the Newton-Raphson method yields

$$u\left.\frac{\partial u}{\partial s}\right|_{i+\frac{1}{2},j} = \left(\frac{\bar{u}_{i+1,j}}{\Delta s}\right) u_{i+1,j} - \frac{1}{2\Delta s}\left(\bar{u}^2_{i+1,j} + u^2_{i,j}\right)$$

and

$$v\left.\frac{\partial u}{\partial n}\right|_{i+\frac{1}{2},j} = \frac{v_{i+\frac{1}{2},j}}{4\Delta n}\left(\bar{u}_{i+1,j+1} - \bar{u}_{i+1,j-1} + u_{i,j+1} - u_{i,j-1}\right)$$

$$+ \frac{\bar{v}_{i+\frac{1}{2},j}}{4\Delta n}\left(u_{i+1,j+1} - u_{i+1,j-1} + u_{i,j+1} - u_{i,j-1}\right)$$

$$- \frac{\bar{v}_{i+\frac{1}{2},j}}{4\Delta n}\left(\bar{u}_{i+1,j+1} - \bar{u}_{i+1,j-1} + u_{i,j+1} - u_{i,j-1}\right) .$$

After solving for the unbarred quantities, the equation may be written as

$$A_1 u_{i+1,j-1} + B_1 u_{i+1,j} + C_1 u_{i+1,j+1} + E_1 v_{i+1/2,j} = D_1$$

where

$$A_1 = -\frac{\bar{v}_{i+1/2,j}}{4\Delta n} - \frac{1}{2\Delta n^2}$$

$$B_1 = \frac{\bar{u}_{i+1,j}}{\Delta s} + \frac{1}{\Delta n^2}$$

$$C_1 = \frac{\bar{v}_{i+1/2,j}}{4\Delta n} - \frac{1}{2\Delta n^2}$$

$$E_1 = (\bar{u}_{i+1,j+1} - \bar{u}_{i+1,j-1} + u_{i,j+1} - u_{i,j-1})/4\Delta n$$

and

$$D_1 = \frac{\bar{u}^2_{i+1,j} + u^2_{i,j}}{2\Delta s} + \bar{v}_{i+1/2,j}\,\frac{(\bar{u}_{i+1,j+1} - \bar{u}_{i+1,j-1})}{4\Delta n}$$

$$+ \frac{u^2_{e_{i+1}} - u^2_{e_i}}{2\Delta s} + \frac{(u_{i,j+1} - 2u_{i,j} + u_{i,j-1})}{2\Delta n^2} \quad .$$

6. Derivation of Potential Solution for Ellipsoid of Revolution

In cartesian coordinates, the velocity components in the x, y, and z directions may be expressed as (Ref. 3)

$$u_x = \left\{ \frac{2u_\infty \cos\alpha}{2 - A_o} \left(\frac{r}{R_2}\right)^2 + \frac{2u_\infty \sin\alpha}{2 - B_o} \left(\frac{x}{a} - 1\right) \frac{r\cos\phi}{R_2} \right\} \Big/ a^2 p^2$$

$$u_y = \left\{ \left(- \frac{2u_\infty \sin\alpha}{2 - B_o} \right) \left[\left(\frac{x}{a} - 1\right)^2 + \frac{r^2 \sin^2\phi}{R_2^2} - \frac{2u_\infty \cos\alpha}{2 - A_o} \left(\frac{x}{a} - 1\right) \right] \right.$$
$$\left. \times \frac{r\cos\phi}{R_2} \right\} \Big/ a^2 p^2$$

and

$$u_z = \left\{ - \frac{2u_\infty \cos\alpha}{2 - A_o} \left(\frac{x}{a} - 1\right) \frac{r\sin\phi}{R_2} + \frac{2u_\infty \sin\alpha}{2 - B_o} \frac{r^2 \sin\phi\cos\phi}{R_2^2} \right\} \Big/ a^2 p^2$$

where p^2 is defined by

$$a^2 p^2 = \left(\frac{x}{a} - 1\right)^2 + \left(\frac{r}{R_2}\right)^2$$

and

$$R_2 = b^2/a$$

(see Figure 7.1). The parameters A_o and B_o are related to the eccentricity and are defined as follows

$$A_o = \frac{2(1 - e^2)}{e^3} \left\{ \frac{1}{2} \ell n \left[\frac{1 + e}{1 - e} \right] - e \right\}$$

and

$$B_o = \frac{1}{e^2} - \frac{(1 - e^2)}{2e^3} \ell n \left[\frac{1 + e}{1 - e} \right]$$

where

$$e = \sqrt{(1 - (b/a)^2)} \quad .$$

In cylindrical coordinates, the velocity components may be expressed as

$$u_r = u_y \cos \phi + u_z \sin \phi$$

and

$$u_\phi = u_z \cos \phi - u_y \sin \phi \quad .$$

Substitution of the appropriate parameters yields

$$u_r = \left\{ - \frac{2u_\infty \cos \alpha}{2 - A_o} \left(\frac{x}{a} - 1 \right) \frac{r}{R_2} - \frac{2u_\infty \sin \alpha}{2 - B_o} \left(\frac{x}{a} - 1 \right)^2 \cos \phi \right\} \Big/ a^2 p^2$$

and

$$u_\phi = \frac{2u_\infty \sin \alpha \sin \phi}{2 - B_o} \quad .$$

The component of the total velocity along a body meridian, u_m, may be obtained from

$$u_m = \sqrt{u_x^2 + u_r^2} \quad .$$

Substitution of the appropriate velocities yields

$$u_m = \left\{ \frac{2u_\infty \cos \alpha}{2 - A_o} \frac{r}{R_2} + \frac{2u_\infty \sin \alpha}{2 - B_o} \left(\frac{x}{a} - 1 \right) \cos \phi \right\} \Big/ \sqrt{a^2 p^2} \quad .$$

The edge velocity, u_e, may be obtained from

$$u_e = \sqrt{u_m^2 + u_\phi^2} \quad .$$

Substitution of the appropriate expressions for the velocity components u_m and u_ϕ then gives

$$u_e^2 = \left\{ \frac{2u_\infty \cos \alpha}{2 - A_o} \frac{r}{R_2} + \frac{2u_\infty \sin \alpha}{2 - B_o} \left(\frac{x}{a} - 1 \right) \cos \phi \right\}^2 \Big/ a^2 p^2$$

$$+ \left\{ \frac{2u_\infty \sin \alpha \sin \phi}{2 - B_o} \right\}^2 \quad .$$

7. Derivation of Analytical Expression for Streamline Angle and Circumferential Derivative for Ellipsoid of Revolution

The streamline angle is related to the circumferential and meridianal velocity components and may be shown to be

$$\tan \theta = \frac{u_\phi}{u_m} \quad .$$

Substitution of the appropriate velocity components (from Appendix A, Section 6) yields

$$\tan \theta = \frac{\sin \alpha \sin \phi \sqrt{\left(\frac{x}{a} - 1\right)^2 + \left(\frac{r}{R_2}\right)^2}}{\left\{\frac{r}{R_2} \cos \alpha \frac{2 - B_o}{2 - A_o} + \sin \alpha \cos \phi \left(\frac{x}{a} - 1\right)\right\}} \quad .$$

Application of the product rule to this equation yields

$$\left.\frac{\partial \theta}{\partial \phi}\right|_x = \left[\frac{\sin \alpha \cos^2 \theta \sqrt{\left(\frac{x}{a} - 1\right)^2 + \left(\frac{r}{R_2}\right)^2}}{\left\{\frac{r}{R_2} \cos \alpha \left(\frac{2 - B_o}{2 - A_o}\right) + \sin \alpha \cos \phi \left(\frac{x}{a} - 1\right)\right\}}\right] \cdot$$

$$\left[\cos \phi + \frac{\sin \alpha \sin^2 \phi \left(\frac{x}{a} - 1\right)}{\left(\frac{r}{R_2} \cos \alpha \left(\frac{2 - B_o}{2 - A_o}\right) + \sin \alpha \cos \phi \left(\frac{x}{a} - 1\right)\right)}\right] \quad .$$

APPENDIX B. INPUT PARAMETERS AND SUBROUTINES

1. Description of Input Parameters

Required inputs to the main program consist of the following parameters (note that all parameters describing the body geometry must be input with the same units).

RPER — radius of spherical cap (dimensionless, ft or m)

XNOSE — distance from body nose to origin of body axes (see Figure 4.1)(dimensionless, ft or m)

BL — body length, L (dimensionless, ft or m)

XINT — axial location of sphere-afterbody interface (see Figure 4.1)(dimensionless, ft or m)

DST — maximum step size along a streamline, Δs (dimensionless, ft or m)

DPSI — step size in degrees of arc on spherical cap, Δx (degrees)

ALPD — effective angle-of-attack, α_{eff} (degrees)

NBS — Boundary Layer Method

0 for Hall's method

1 for Blottner's method

NT — Edge Test

0 for point added at edge when velocity at point next to edge is less than 99.95% of point at edge

1 for point added when $(C_f \sqrt{R_{e_L}})_e$ is above 0.005

NC — Convergence Test

0 for convergence based on $(C_f \sqrt{R_{e_L}})_{(i)}$ changing by less than 0.5% between successive iterations

1 for convergence based on each point in profile changing by less than 0.1%

MOT — Method to Integrate Streamlines

1 Milne's predictor-corrector method

2 Runge-Kutta method

3 Gear's method for stiff system of differential equations

MAXS — number of stations to be computed

KP — N for velocity profiles printed every Nth station

0 for no velocity profiles printed

KPH — 1 for iterative profiles printed after each Nth station

0 for no iteration profiles printed

KPO — Print Out Type

0 for ordinary print out

1 for additional print out

KBM — number of streamlines to compute

KBMS — Indicator for Streamline Shifting

0 for ordinary run in which a circumferential position at the interface is specified

1 for a streamline shift to be made from the windward streamline to a circumferential position of 1 degree at XMAX (which will be input as PHIPD)

PHIPD circumferential position at interface (degrees) (for KBMS = 0)

XMAX axial position at which the streamline shift is to be made (for KBMS = 1)(dimensionless, ft or m)

ISO Type of Integration to be Performed
0 for integration of both boundary layer and streamlines
1 for integration of streamlines only.

Subroutine PRESS reads in the number of axial and circumferential pressure stations as well as the pressure data (which must be in the form of a pressure coefficient). The pressure data is read in one complete axial station at a time. The parameters relevant to this subroutine are also shared with the pressure fitting routine and are as follows:

NCS number of circumferential stations

NAS number of axial stations

PHI(j) array of circumferential pressure stations (NCS values to be input)(degrees)

X(j) array of axial pressure stations (NAS values to be input)(dimensionless, ft or m)

CP(i,j) pressure coefficient data

i = (1, NAS), j = (1, NCS) .

The axial pressure derivatives (at the interface and body end) for each circumferential plane are read in two at a time.

CPX(1,j) axial derivatives at interface and body end
CPX(2,j) j = (1, NCS) .

2. Subroutine BGEOM

Subroutine BGEOM computes the geometric properties relative to the body axes used in the streamline and boundary layer calculations. For an input axial position, subroutine BGEOM computes the body radius and its derivative, and the body angle Γ and its derivative (see Figure 2.2).

A call to subroutine BGEOM has the form

```
CALL BGEOM (X,R,DRDX,GM,DGX)
```

where the input argument is

X	axial location, x

and the output arguments are

R	body radius, r
DRDX	dr/dx
GM	angle $\Gamma = \tan^{-1}(dr/dx)$
DGX	$d\Gamma/dx$.

3. Subroutine BLOTNR

Subroutine BLOTNR evaluates the boundary layer parameters for use in Blottner's method. These parameters consist of ξ, the transformed coordinate along the streamline and $\bar{\beta}$ the pressure gradient term.

On the spherical nose, the expression for ξ is integrated by the Runge-Kutta method to yield ξ_{i+1} and $\xi_{i+\frac{1}{2}}$ (see BOUNDARY LAYER METHODS). On the afterbody, ξ_{i+1} is evaluated by the integration routine used in the main program and then becomes an input to the subroutine.

A call to subroutine BLOTNR has the form

```
CALL BLOTNR (PSI,S,Y,F,DEXI,EXIH,BETA,DST)
```

where the input arguments are

PSI — position on spherical cap in radians of arc length, ψ

S — distance along streamline (used only on afterbody), S

Y(j) — array of dependent variables (j = 1,6)

F(j) — array of first derivatives of dependent variables (j = 1,6), $F = \frac{dY}{dS}$

DEXI — transformed step size along streamline, $\Delta\xi = \xi_{i+1} - \xi_i$

DST — step size along streamline, Δs

and the output arguments are

EXIH — transformed coordinate at mid-point of interval, $\xi_{i+\frac{1}{2}}$

BETA — pressure gradient parameter, $\bar{\beta}$.

4. Subroutine FCN

Subroutine FCN computes the first derivative for each of the dependent variables to be used in one of the streamline integration routines.

A call to subroutine FCN has the form

```
CALL FCN (N,S,Y,F)
```

where the input arguments are

N	number of differential equations
S	distance along streamline (independent variable), S
Y(j)	array of dependent variables (j = 1,N)

and the output argument is

F(j)	array of first derivatives (j = 1,N), $F = \frac{dY}{dx}$.

5. Subroutine COEFF

Subroutine COEFF calculates the coefficients to the appropriate finite-difference boundary-layer equations. These coefficients are functions of both the normal and tangential grid spacing, the pressure gradient, and the velocity profile at the previous computational station (see BOUNDARY LAYER METHODS).

A call to subroutine COEFF has the form

```
CALL COEFF (JMAX,AM,BM,CM,DM,AS,CS,DS,W,WL,KB)
```

where the input arguments are

JMAX — number of grid locations in normal direction

W(j) — matrix of present iterative values of transformed velocity components (j = 1,JMAX)(dimensionless), $u_{i+1,j}$

WL(j) — matrix of transformed velocity components at last integration station (j = 1,JMAX)(dimensionless), $u_{i,j}$

KB — indicator variable

1 for calculation of coefficients of Blottner's similarity equations at stagnation point

2 for calculation of coefficients of Blottner's non-similar equations

3 for calculation of coefficients of Hall's equations

and the output arguments are

AM(j), BM(j), CM(j), DM(j), AS(j), CS, DS(j)	coefficients of respective boundary layer equations

(all arrays are of dimension JMAX)

6. Subroutine INVERT

Subroutine INVERT solves a block tridiagonal system of linear equations using the modified Davis algorithm. The coupled continuity and momentum equations form such a system.

A call to subroutine INVERT has the form

```
CALL INVERT (JMAX,A,B,C,D,AS,CS,DS,W,KB)
```

where the input arguments are

JMAX	number of normal grid points
A(j) B(j) C(j) D(j) AS(j) CS DS(j)	coefficients to respective boundary layer equations (all arrays are of dimension JMAX)
KB	indicator variable 1 for Blottner's method at stagnation point 2 for Blottner's method at all other points 3 for Hall's method

and the output argument is

W(j) array of grid velocities (dimensionless) (j = 1,JMAX), $u_{i+1,j}$.

7. Function KRUNGE

Function KRUNGE is a subprogram which uses the fourth-order Runge-Kutta method to integrate a system of NDE first-order, ordinary differential equations with a variable step size. As a criterion for varying the computing interval, the differential equations are integrated over an interval of step size DSS first and then over the same interval with two step sizes of DSS/2. The two solutions are then compared to give an estimate of the error for each variable. If any error is larger than EPS = E - 04, these answers are discarded and the computing interval H is halved. If all of the error estimates are less than EPS, the answers are allowed and the integration process continues. In addition, the step size is either doubled or set equal to DST, whichever is the smaller, for the next integration cycle.

The function RUNGE is used in the main program and has the form

```
K = KRUNGE (Y,F,S,DSS,NDE,DST,MR)
```

where the input arguments are

Y(j) — array of dependent variables to be integrated, (j = 1,NDE)

F(j) — array of first derivatives of the dependent variables, (j = 1,NDE), F = dY/dS

S — independent variable (distance along a streamline)

DSS integration step size

NDE number of differential equations

DST maximum integration step size, ΔS

MR indicator variable, MR = 1 for the previous integration interval to be recomputed with a new step size DSS determined in the main program

and the output arguments are

Y(j) array of updated dependent variables (j = 1,NDE)

F(j) array of updated derivatives (j = 1,NDE)

K indicator variable

0 implies completion of integration cycle

1 implies the integration cycle has not been completed

2 implies the step size has been reduced to a value below E-08 .

8. Subroutine MILNES

Subroutine MILNES uses the fourth-order predictor-corrector method of Milnes to numerically integrate a system of NDE first-order, ordinary differential equations. Since this method is not self-starting, it must be used in conjunction with an alternate method (such as the Runge-Kutta or Gear method) to generate the starting values. The advantage of a method of this type is that the computational work is kept to a minimum between integration steps. This, however, is achieved at the price of accuracy since the step size is held constant over the entire interval regardless of the error introduced.

A call to subroutine MILNES has the form

```
CALL MILNES (Y,F,PCM,NDE,DST,S)
```

where the input arguments are

Y(j)	array of dependent variables (j = 1,NDE)
F(j)	array of first derivatives of dependent variables (j = 1,NDE), $F = \frac{dY}{dS}$
PCM	temporary storage of both the dependent variables and their first derivatives at four previous stations
NDE	number of differential equations
DST	step size along streamline, ΔS

S — streamline distance (independent variable), S

and the output arguments are

Y(j) — updated array of dependent variables at next integration step, (j = 1,NDE)

F(j) — updated array of first derivatives, (j = 1,NDE)

S — updated value of independent variable, S .

9. Subroutine DGEAR

Subroutine DGEAR integrates a system of first-order differential equations using the backward differentiation formulas of Gear (Ref. 12). This technique is particularly well suited to situations in which the system of differential equations may be classified as stiff. In these types of applications, other techniques would be apt to decrease the integration step size to prohibitively small values in an attempt to satisfy the allowable error tolerance. Gear's method, however, has the property of "stiff stability" which effectively removes the limitations on the step size. The integration step size is adjusted in the routine so as to satisfy the error tolerance specified by the user. The technique used is similar to that employed in the Runge-Kutta method described in Function KRUNGE. The method also necessitates that (in general) a nonlinear system of algebraic equations be solved at each step of the integration. To solve these equations, the integration package has the option of employing a variety of iterative schemes.

A call to subroutine DGEAR has the form

```
CALL DGEAR (NDE,FCN,FCNJ,S,HG,Y,SEND,TOL,METH,MITER,INDEX,IWK,WK,IER)
```

where the input arguments are

NDE	number of differential equations to be integrated
FCN	subroutine to evaluate first derivatives of differential equations

FCNJ — subroutine to evaluate the Jacobian of the system of differential equations -- this parameter may or may not be specified depending on other quantities specified in the argument

S — distance along streamline, S

HG — integration step size along streamline

Y(j) — array of dependent variables at present station (j = 1,NDE)

SEND — value of independent variable at which dependent variables are desired, S + ΔS

TOL — maximum error tolerance allowed between integration steps

METH — Basic Integration Method

1 for use of Adam's method

2 for use of Gear's method

MITER — Iteration Method

0 for functional iteration, internal calculation of the Jacobian

1 for chord method, Jacobian is supplied externally

2 for chord method, internal calculation of Jacobian

3 for chord method, diagonal approximation of Jacobian is made internally

INDEX 1 for first call to subroutine

0 for remaining calls

IWK work vector of length NDE

WK work vector of length 13 x NDE

IER error parameter

33 for error test not satisfied due to too low an error tolerance

66 for error test was satisfied only after HG was reduced

132 error test failed after HG was decreased to lower limit .

10. Subroutine PRESS

Subroutine PRESS reads in the pressure data used in the pressure fitting technique. The input pressures must be in the form of a pressure coefficient. The data read into the subroutine consist of the following:

NCS	number of circumferential pressure stations
NAS	number of axial pressure stations
PHI(j)	array of circumferential stations (degrees) (j = 1,NCS)
X(i)	array of axial stations (i = 1,NAS)
CP(i,j)	pressure coefficient data (i = 1,NAS, j = 1,NCS)
CPX(i,j)	axial pressure derivatives at body interface and end, (i = 1,2, j = 1,NCS) .

11. Subroutine SPHCAP

Subroutine SPHCAP computes the inviscid flow properties on the spherical nose of the body to be used in the boundary layer integration. On the initial call to the subroutine, a check in continuity between the pressure and its axial derivative across the interface is first made. If the values of the parameters should vary by more than 0.003 or 4.0, respectively, the quadratic spline technique is extended to the stagnation point. Otherwise, the series expression for the pressure coefficient will be used. This in turn will yield the dimensionless fluid velocity which is used in the subsequent boundary layer calculations. Additionally, the derivative of the pressure coefficient with respect to ψ is calculated for use in Blottner's boundary layer method.

A call to subroutine SPHCAP has the form

```
CALL SPHCAP (P,CP,UE,DCPSI)
```

where the input argument is

P	position on nose in radians of arc length form stagnation point, ψ

and the output arguments are

CP	pressure coefficient, C_p
UE	fluid velocity at edge of boundary layer (dimensionless), u_e

DCPSI derivative of pressure coefficient with respect to ψ, $\frac{\partial c_p}{d\psi}$.

an additional input to the subroutine is

COE(i) coefficients used in the series expression for CP ($i = 1,10$) and is specified in a data statement within the subroutine (see SURFACE PRESSURE DISTRIBUTION).

12. Subroutine MIDPTS

Subroutine MIDPTS calculates the dependent variable at the midpoint at each of the subintervals defined within the parabolic spline technique (see SURFACE PRESSURE DISTRIBUTION). This amounts to solving a tridiagonal system of linear algebraic equations by the LU decomposition method.

A call to subroutine MIDPTS has the form

```
CALL MIDPTS (AA,BB,CC,D,YM,N)
```

where the input arguments are

AA(i), BB(i), CC(i), D(i)	arrays of coefficients of tridiagonal system (i = 1, N - 1)
N	number of discrete data points along interval to be spline fit

and the output argument is

YM(i)	array of dependent variables at midpoints (i = 1, N -1), Y .

13. Subroutine INVISD

Subroutine INVISD computes the properties of the inviscid flow field at any point on the body surface after the interface. These properties consist of the pressure coefficient, the local fluid velocity at the edge of the boundary layer, the first and second circumferential pressure coefficient derivatives, the axial pressure coefficient derivative and the mixed derivative. These parameters result from a quadratic spline fit to the input pressure data (see SURFACE PRESSURE DISTRIBUTION). The method employed is limited in application to situations in which the number and position of the circumferential stations do not vary between axial stations.

A call to subroutine INVISD has the form

```
CALL INVISD (XX,PPH,UE,DCPX,DCPPH,CPC,DCPXP,D2PPH)
```

where the input arguments are

XX — axial position, x

PPH — circumferential position (radians), ϕ

and the output arguments are

UE — fluid velocity at edge of boundary layer (dimensionless), u_e

DCPX — axial derivative of pressure coefficient, $\frac{\partial C_p}{\partial x}$

DCPPH circumferential derivative of pressure coefficient, $\frac{\partial C_p}{\partial \phi}$

CPC pressure coefficient, C_p

DCPXP mixed axial-circumferential second derivative of pressure coefficient, $\frac{\partial^2 C_p}{\partial x \partial \phi}$

D2PPH second circumferential derivative of pressure coefficient, $\frac{\partial^2 C_p}{\partial \phi^2}$

other inputs to the subroutine consist of

NAS number of axial pressure stations

NCS number of circumferential pressure stations

X(i) array of axial pressure stations (i = 1,NAS)

PHI(j) array of circumferential pressure stations (degrees), (j = 1, NCS)

CP(i,j) pressure coefficient data (i = 1,NAS , j = 1,NCS)

CPX(i,j) axial pressure derivatives at body interface and end (i = 1,2 , j = 1,NCS) .

14. Subroutine STAGN

Subroutine STAGN locates the stagnation point on a given configuration. For the case of the sphere-ogive-cylinder geometry, this amounts to calculating the Newtonian stagnation given an effective angle of attack. For configurations having analytical pressure distributions, the stagnation point may be calculated from analytical expressions which must be supplied by the user (as was the case of the ellipsoid of revolution).

A call to subroutine STAGN has the form

```
CALL STAGN (ALP,XO,XNOSE,RPER)
```

where the input arguments are

ALP	effective angle of attack (radians), α_{eff}
XNOSE	distance from origin of body axes to body nose
RPER	radius of spherical cap, R_{per}

and the output argument is

XO	axial location of stagnation point .

15. Listing of Main Program

```
      DIMENSION AM(50),BM(50),CM(50),DM(50)
      DIMENSION WS(50), W(2,50), WL(2,50), WO(40)
      DIMENSION AS(50),DS(50)
      DIMENSION Y(6),F(6),PCM(6,4,2),IWK(6),WK(120)
      EXTERNAL FCN,FCNJ
      COMMON /HALL/DSTN,DY,UE
      COMMON /BLCT/DETA,DEXI,EXIM,BIPH
      COMMON /SCALE/HL,HLH,H
      COMMON /INTFC/XINT,PHIP,SAL,CAL,PSIMAX,PSIO,RPER
      COMMON /CPSIA/NAS,NCS
      COMMON /OUTPT/R,CP,DCPX,DCPPH,DCPXP,D2PPH,GM,DGX
C
C
C
C
C
C     DESCRIPTION OF INPUT PARAMETERS:
C
C
C  RPER      RADIUS OF SPHERICAL CAP
C  XNOSE     DISTANCE FROM ORIGIN OF BODY AXES TO NOSE OF BODY
C  BL        TOTAL BODY LENGTH
C  XINT      AXIAL LOCATION OF SPHERE-AFTERBODY INTERFACE
C  DST       INTEGRATION STEP SIZE ALONG STREAMLINE
C  DPSI      STEP SIZE ON SPHERICAL CAP (DEGREES OF ARC LENGTH)
C  ALPD      EFFECTIVE ANGLE OF ATTACK (DEGREES)
C
C  NBS       0 FOR HALL'S BOUNDARY LAYER METHOD
C            1 FOR BLOTTNER'S METHOD
C  NT        EDGE TEST: 0 FOR POINT ADDED AT EDGE WHEN VELOCITY
C                       AT POINT
C                       NEXT TO EDGE IS LESS THAN 99.95 PERCENT OF
C                       POINT AT EDGE.
C                     1 FOR POINT ADDED WHEN EDGE SKIN FRICTION
C                       COEFFICIENT  IS ABOVE 0.0005.
C  NC        CONVERGENCE TEST: 0 FOR CONVERGENCE BASED ON CFREX
C                              CHANGING BY LESS THAN 0.5 PERCENT
C                              BETWEEN SUCCESSIVE ITERATIONS.
C                            1 EACH POINT IN PROFILE IS CHANGING BY
C                              LESS THAN 0.1 PERCENT.
C  MOT       METHOD USED TO INTEGRATE STREAMLINES:
C            1 FOR MILNE'S PREDICTOR CORRECTOR METHOD
C            2 FOR RUNGE-KUTTA METHOD
C            3 FOR GEAR'S METHOD (STIFF SYSTEM)
C  MAXS      NUMBER OF STATIONS TO BE COMPUTED
C  KP=N      FOR PROFILES PRINTED EVERY NTH STATION
C  KPH       0 FOR NO INTERATION PROFILES PRINTED
C            1 FOR ITERATION PROFILES PRINTED
C  KPO       0 FOR ORDINARY PRINT OUT
C            1 FOR ADDITIONAL PRINT OUT
C  KBM       NUMBER OF STREAMLINES TO TRACE
C  KBMS      0 FOR ORDINARY RUN
C            1 FOR STREAMLINE SHIFT FROM PHI=0 TO PHI=1.0 AT XMAX
C  PHIPD     CIRCUMFERENTIAL ANGLE AT INTERFACE (KBMS=0) (DEGREES)
C  XMAX      POSITION ON WINDWARD STREAMLINE WHERE STREAMLINE
C            SHIFT FROM WINDWARD PLANE IS MADE (KBMS=1)
```

```
C
C  ISO       1 FOR INTEGRATION OF STREAMLINES ONLY
C            0 FOR ADDITIONAL INTEGRATION OF BOUNDARY LAYER
C
C
C
C    STREAMLINE INTEGRATION:
C
C  S = STREAMLINE DISTANCE (INDEPENDENT VARIABLE)
C  Y(1)   AXIAL POSITION
C  Y(2)   CIRCUMFERENTIAL POSITION
C  Y(3)   STREAMLINE ANGLE
C  Y(4)   D(THETA)/D(PHI)
C  Y(5)   LOG(D(PHI)/D(BETA))
C  Y(6)   EXI  (FOR BLOTTNER'S METHOD ONLY)
C
C
C
C
      PI=ACOS(-1.)
      DGR=PI/180.
C
C
      READ(1,2) FREN,XNOSE,BL,XINT,DST,DPSID,ALPD
   2  FORMAT(4(1X,F10.5),/,3(1X,F10.5))
      READ(1,4) NBS,NT,NC,MOT,MAXS,KP,KPH,KPO,KBM,KBMS,ISO
   4  FORMAT(11(1X,I3))
      JMAX=40
      ETAE=4.5
      YEDGE=0.15
      CALL PRESS
      MDE=5+NBS
      IF(ISO.EQ.1) GO TO 36
  46  W(1,1)=0.0
      W(1,JMAX)=1.0
      W(2,1)=0.0
      WL(1,1)=0.0
      WL(1,JMAX)=1.0
      WL(2,1)=0.0
      DETA=ETAE/(JMAX-1)
      ITER=0
      CHKEXI=1.0
      DY=YEDGE/(JMAX-1)
      DYS=DY
      DSTS=DST
C
C
C
C
C
C  GUESS INITIAL PROFILE
      DO 10 J=2, JMAX
      W(1,J)=1.0
      WL(1,J)=1.0
      WS(J)=1.0
      W(2,J)=W(2,J-1)-0.5*DETA*(W(1,J)+W(1,J-1))
  10  CONTINUE
C
C
C
      KB=1
```

```
 30   CONTINUE
      CALL COEFF(JMAX,AM,BM,CM,DM,AS,CS,DS,W,WL,KB)
      CALL INVERT(JMAX,AM,BM,CM,DM,AS,CS,DS,W,KB)
1090  FORMAT(/,10(2X,F8.6))
      ITER=ITER+1
      KB=2
      IF(NC.EQ.0) GO TO 32
      DO 34 J=2, JMAX
      IF(ABS((WS(J)-W(1,J))/W(1,J)).GE.0.001) KB=1
      WL(1,J)=W(1,J)
 34   WS(J)=W(1,J)
      IF(KB.EQ.1) GO TO 30
 32   FP=(2.*W(1,2)-0.5*W(1,3))/DETA
      CFREX=2.*SQRT(2.)*FP
      CFREXC=CFREX
      IF(NC.EQ.1) GO TO 36
      IF((ABS(CFREX-CFREXL)/CFREXL).GT.0.005) KB=1
      CFREXL=CFREX
      IF(KB.EQ.1) GO TO 30
      FPE=-(2.*W(1,JMAX-1)-0.5*W(1,JMAX-2)-1.5*W(1,JMAX))/DETA
      CFREXE=2.*SQRT(2.)*FPE
      IF(CFREXE.LE.0.002) GO TO 36
      ETAE=ETAE+DETA
      GO TO 46
 36   WRITE(3,5)
 5    FORMAT(//,20X,'INPUT PARAMETERS: ',/)
      WRITE(3,3) RPER,XNOSE,BL,XINT,DST,DPSID,ALPD,MAXS,NT,NC,NOT,
     *NBS,KP,KPH,KPC,NCS,NAS
 3    FORMAT(/,23X,'RPER = ',8X,F8.5,5X,'X(NOSE) = ',5X,F8.5,/,23X,
     *'BODY LENGTH = ',1X,F8.5,5X,'X(INTERFACE) = ',F8.5,/,23X,
     *'DST  = ',8X,F8.5,5X,'DELTA PSI = ',3X,F8.5,/,23X,
     *'ANG. ATTACK = ',1X,F8.5,5X,'NO. STATIONS = ',I8,/,23X,
     *'EDGE TEST = ',3X,I2,3X,'CONV. TEST =',2X,I2,3X,'INTEG. METH = '
     *,1X,I2,/,23X,'B.L. METHOD = ',1X,I2,3X,'NO. PROFILES = ',I2,3X,
     *'ITER. PROFILES =',I3,/,23X,'EXTRA P.O. = ',2X,I2,3X,
     */,23X,'NO. AXIAL PRESSURE STATIONS = ',I3,/,23X,
     *'NO. CIRCUMFERENTIAL PRESSURE STATIONS = ',I3,//)
      IF(NBS.EQ.0.OR.KBM.EQ.1.OR.ISO.EQ.1) GO TO 45
      DO 45 J=1,JMAX
      WO(J)=W(1,J)
 45   CONTINUE
      DETAS=DETA
      ALP=ALPD*DGR
      CAL=COS(ALP)
      SAL=SIN(ALP)
      YC=0.0
      GMC=90.*DGR
      CALL STAGN(ALP,XO,XNOSE,RPER)
      IF(XO.NE.0.0) CALL DGEOM(XO,YO,DRDX,GMO,DGX)
      CGC=COS(GMC)
      SGC=SIN(GMC)
      WRITE(3,7) XO,YO
 7    FORMAT(/,23X,'STAG. POINT, XO =',F8.5,/,23X,'STAG. POINT, YO =',
     *F8.5,/)
      IF(KP.EQ.1) WRITE(3,1090) (W(1,J),J=1,JMAX)
      WRITE(3,9) CFREX,ITER
 9    FORMAT(/,10X,'STAGNATION POINT CFREX = ',F9.6,2X,
     *'CONVERGENCE IN ',I3,1X,'ITERATIONS',/)
      PSIC=ACOS((RPER-XINT+XNOSE)/RPER)-ALP
      IF(PSIC.GT.DGR) PSIC=DGR
      PSICD=PSIC/DGR
```

```
      DGR=SIN(PSIC)
      DEPS=DCF*BPER
      CALL BGECM(XINT,RI,DRI,GMI,DGI)
      SGI=SIN(GMI)
      CGI=COS(GMI)
      WRITE(3,33) DGR,PSIDD,DPSID
   33 FORMAT(/,10X,'DEPS/BPER = ',F14.6,/,10X,'PSI(EPS) =',F7.3,/,10X,
     *'DPSI = ',F5.2,/)
C
C
      KBS=1
    8 READ(1,6) PHIPD
    6 FORMAT(F10.5)
      IF(KEMS.EQ.0) GO TO 12
      XMAX=PHIPD
      PHIPD=0.0
   12 CONTINUE
      JMAX=40
      IF(KBS.EQ.1.OR.NBS.EQ.0) GO TO 13
      DO 13 J=1,JMAX
      WL(1,J)=WC(J)
   13 CONTINUE
      LK=0
      HL=0.
      KI=0
      NHP=0
      CPL=CPBEXC
      S=BPER*PSIC
      SL=0.0
      PHIP=PHIPD*DGR
      DENSQ=(1.+TAN(GMI)**2)*SIN(PHIP)**2*TAN(ALP)**2
     ++(1.+TAN(GMI)*COS(PHIP)*TAN(ALP))**2
      BETA=ACOS((COS(PHIP)+TAN(GMI)*TAN(ALP))/SQRT(DENSQ))
      BETAD=BETA/DGR
      PSIMAX=ACOS(CAL*SGI+SAL*CGI*COS(PHIP))
      PSIMD=PSIMAX/DGR
      DPSI=DPSID*DGR
      PSI=PSIC
      IF(PSI.GE.PSIMAX) PSI=PSIMAX
      NBPS=(PSIMAX-PSIC)/DPSI
      WRITE(3,51) PHIPD,BETAD,PSIMD
   51 FORMAT(/,10X,'PHI(INTERFACE) = ',F6.2,/,10X,'BETA = ',F7.3,/,
     *10X,'PSI(MAX) = ',F8.4,//)
      CALL SPHCAP(0.0,CP,UE,DCPSI)
      IF(ISC.EQ.0) GO TO 53
      H=BPER*SIN(PSIMAX)
      HL=H
      S=BPER*PSIMAX
      GO TO 42
   53 CONTINUE
      WRITE(3,35)
   35 FORMAT(//,10X,'SPHERICAL CAP RESULTS: ',/)
      WRITE(3,52)
   52 FORMAT(6X,'PSI',13X,'CP',14X,'UE/VIN',10X,'CPREX',11X,'H',15X,
     *'S',15X,'ITER',12X,'CPREX(EDGE)',/)
      DPR=DPSI*BPER
      IF(NBS.EQ.1) GO TO 39
      CALL SPHCAP(PSI,CP,UE,DCPSI)
      DO 60 J=2,JMAX
      WL(1,J)=0.0
      W(2,J)=0.0
```

```
      W(1,J)=UE
   60 CONTINUE
      GO TO 41
   39 CALL SEHCAP(PSI,CP,UE,DCPSI)
   41 H=BETA*SIN(PSI)
      NHE=NHE+1
      GO TO 600
   40 PSID=PSI/DGR
      WRITE(3,37) PSID,CP,UE,CFREX,H,S,ITER,CFREXE
   37 FORMAT(/,2X,6(3X,F12.5),11X,I2,3X,F12.5)
      IF(KL.EQ.1) GO TO 42
      IF(NHE.LT.NHES) GO TO 38
      S=BETA*PSIMAX
      PSI=PSIMAX
      KL=1
      GO TO 39
   38 S=S+DPS
      PSI=PSI+DPSI
      GO TO 39
C
C
C
C
C
C
C
C
C
C
C
C
C
C
C
   42 LK=1
      HG=0.00001
      TOL=0.00001
      MITER=3
      INDEX=1
      METH=2
      WRITE(3,1040)
 1040 FORMAT(//,6X,'X',15X,'PHI',13X,'THETA',11X,'S ',14X,
     *'H',15X,'UE/VIN',10X,'DPDS',12X,'CFREL')
      IF(KEC.EQ.1) WRITE(3,1042)
 1042 FORMAT(6X,'CP',14X,'DCP/DX'10X,'DCP/DPHI',8X,'DCPXP',11X,
     A'D2PPH',11X,'S',15X,'ITER  ',10X,'CFREL(EDGE)',/,
     A6X,'D(THETA)/DPHI',3X,'LOG(DPHI/DPETA)')
 1050 FORMAT(//,2X,4(3X,F12.5),1X,E14.6,3(3X,F12.5))
 1052 FORMAT(2X,6(3X,F12.5),3X,I10,5X,F12.5,/,2X,2(3X,F12.5))
C
C
C
C  BEGIN TO TRACE STREAMLINE OVER BODY,  AFTER EACH INTEGRATION STEP
C  COMPUTE THE BOUNDARY LAYER USING EITHER BLOTTNERS OR HALLS METHOD
C
C
C
C
C  INITIAL VALUES AT INTERFACE
      DST=DSTS
      DY=DYS
```

```
      DETA=DETAS
C
      Y(1)=XINT
      SPSI=SIN(PSIMAX)
      CPSI=COS(PSIMAX)
      Y(2)=PHIP
      Y(3)=ACOS((CAL*CGI-SAL*SGI*COS(Y(2)))/SPSI)
      IF(Y(2).LE.0.0) Y(3)=0.0
      DPSIDP=(SAL*CGI*SIN(Y(2))/SPSI)
      IF(Y(2).LE.0.0) GO TO 43
      Y(4)=(SAL*SGI*SIN(Y(2))-COS(Y(3))*CPSI*DPSIDP)/
     A (-SPSI*SIN(Y(3)))
   43 IF(Y(1).LE.0.0) Y(4)=SQRT(-SAL*SGI/SPSI+CPSI*SAL*CGI/SPSI**2)
      Y(5)=ALOG(BEER*SPSI/(RI*COS(Y(3))))
      IF(NES.EQ.0) GO TO 44
      CALL FCN(NDE,S,Y,F)
      CALL DICTNR(PSI,S,Y,F,DEXI,EXIH,DIPH,DST)
   44 K=0
      LP=1
      JJ=3
      DSS=DST
      KL=0
      IF(ISC.EQ.1) GO TO 130
      IF(MOT.NE.1) GO TO 550
      DO 307 J=1, NDE
      PCM(J,4,1)=Y(J)
  307 PCM(J,4,2)=0.
C
C
C
  550 SEND=S+DST
      IF(KEMS.EQ.0) GO TO 560
      IF(Y(1).LT.XMAX) GO TO 560
      Y(2)=DGR
      Y(3)=Y(4)*DGR
      INDEX=1
      HG=0.00001
      CALL FCN(NDE,S,Y,F)
      XMAX=RL
  560 CONTINUE
      NRE=NRE+1
      GO TO(305,310,315),MOT
  305 IF(JJ.GE.1) GO TO 315
  309 CALL MILNES(Y,F,PCM,NDE,DST,S)
      GO TO 555
  308 DO 580 J=1,NDE
      PCM(J,JJ,1)=Y(J)
  580 PCM(J,JJ,2)=F(J)
      JJ=JJ-1
      GO TO 555
C
C
C
C
C
  310 K=RKUNGE(Y,F,S,DSS,NDE,DST,MR)
      IF(K.EQ.2) GO TO 142
      CALL FCN(NDE,S,Y,F)
      IF(K.EQ.1) GO TO 310
      IF(S.GE.SEND.OR.KL.EQ.1) GO TO 311
      SP=S
```

```
      GO TO 310
  311 IF(KL.EQ.1) GO TO 312
      MR=1
      DSS=DSS*(SEND-SP)/(S-SP)
      KL=1
      GO TO 310
  312 KL=0
      GO TO 555
C
C
C
  315 CALL DGEAR(NDE,FCN,FCNJ,S,HG,Y,SEND,TOL,METH,MITER,INDEX,IWK,
     *WK,IER)
      CALL FCN(NDE,S,Y,F)
      IF(MOT.EQ.1) GO TO 308
C
C
C
C
  555 H=R*COS(Y(3))*EXP(Y(5))
      IF(ISO.EQ.1) GO TO 130
C
  600 LP=1
      IF(KP.NE.0) LP=MOD(NHP,KP)
C
C
      IF(NBS.EQ.0) GO TO  85
  300 ITER=0
      KB=2
      CALL ELCTNS(PSI,S,Y,F,DEXI,EXIH,BIPH,DST)
C     WRITE(3,5000) Y(6),EXIH,DEXI,BIPH
 5000 FORMAT(/,4X,'EXI  =',E14.6,/,4X,'EXIH =',E14.6,/,4X,'DEXI =',
     *E14.6,/,4X,'BETA =',F14.6,//)
  361 ITER=0
  360 CALL COEFF(JMAX,AM,BM,CM,DM,AS,CS,DS,W,WL,KB)
      CALL INVERT(JMAX,AM,BM,CM,DM,AS,CS,DS,W,KB)
      IF(LP.EQ.0.AND.KPH.EQ.1) WRITE(3,1090) (W(1,J), J=1,JMAX)
      ITER=ITER+1
      IF(ITER.GE.40) GO TO 135
      IF(NC.EQ.0) GO TO 340
      KB=3
      DO 350 J=2, JMAX
      IF(ABS((WS(J)-W(1,J))/W(1,J)).GE.0.001) KB=2
  350 WS(J)=W(1,J)
      IF(KB.EQ.2) GO TO 360
  340 FP=(2.*W(1,2)-0.5*W(1,3))/DETA
      CFREX=SQRT(2.)*FP*SQRT(BL/Y(6))*UE**2*H
      IF(NC.EQ.1) GO TO 370
      KB=3
      IF((ABS(CFREXL-CFREX)/CFREXL).GT.0.005) KB=2
      CFREXL=CFREX
      IF(KB.EQ.2) GO TO 360
  370 FPE=-(2.*W(1,JMAX-1)-0.5*W(1,JMAX-2)-1.5*W(1,JMAX))/DETA
      CFREXE=CFREX*FPE/FP
      IF(N1.EQ.0) GO TO 375
      IF(CFREXE.GE.0.0005) GO TO 380
      GO TO 118
  375 IF((W(1,JMAX-1)/W(1,JMAX)).LT.0.9995) GO TO 380
      GO TO 118
  380 JMAX=JMAX+1
      IF(JMAX.GT.90) GO TO 362
```

```
      WL(1,JMAX)=WL(1,JMAX-1)
      DO 390 J=2,JMAX
  390 W(1,J)=WL(1,J)
      ETAE=ETAE+DETA
      WRITE(3,1030)
      KB=2
C     LE=0
      GO TO 361
  362 JMAX=26
      DO 363 J=1,25
      JX=2*J-1
  363 WL(1,J)=WI(1,JX)
      WL(1,26)=WL(1,50)
      ETAE=ETAE+DETA
      DETA=DETA*2.
      DPSI=DPSI*2.0
      DST=DST*2.0
      DPB=DPB*2.0
      WRITE(3,119) JMAX, DETA
      DO 364 J=2,JMAX
  364 W(1,J)=WL(1,J)
      KB=2
      GO TO 361
C
C
C
C
C
C
C
C
C
C
C
C
C
C
C
C
C  APPLY HALL'S TRANSFORMATION
C  D(X)=D(X/L), D(Y)=D(SQRT(RE)*Y/L), U=U/VIN, UE=UE/VIN
   85 KB=3
      DSTN=(S-SL)/BL
      SN=S/BL
      SH=SL+(S-SL)/2.0
      SHN=SH/BL
      BLH=(B+BL)/2.
      IF(WL(1,JMAX).LE.0.) GO TO 88
      DO 88 J=2, JMAX
      W(1,J)=W(1,J)*UE/WL(1,JMAX)
   88 CONTINUE
   86 ITER=0
   97 CALL COEFF(JMAX,AM,BM,CM,DM,AS,CS,DS,W,WL,KB)
      CALL INVERT(JMAX,AM,BM,CM,DM,AS,CS,DS,W,KB)
      IF(LE.EQ.0.AND.KPR.EQ.1) WRITE(3,1090) (W(1,J),J=1,JMAX)
      ITER=ITER+1
      IF(ITER.GE.40) GO TO 135
      KB=4
      IF(NC.EQ.0) GO TO 196
      DO 182 J=2, JMAX
      IF(ABS((WS(J)-W(1,J))/W(1,J)).GE.0.001) KB=3
  182 WS(J)=W(1,J)
```

```
      IF(KB.EQ.3) GO TO 97
  196 FP=(2.*W(1,2)-0.5*W(1,3))/DY
      CFREX=2.*FP
      IF(NC.EQ.1) GO TO 198
      IF((ABS(CFREXL-CFREX)/CFREXL).GT.0.005) KB=3
      CFREXL=CFREX
      IF(KB.EQ.3) GO TO 97
  198 KB=3
      GO TO 115
  150 JMAX=JMAX+1
      IF(JMAX.LE.50) GO TO 151
      JMAX=26
      DO 116 J=1, 25
      JX=2*J -1
  116 WL(1,J)=WL(1,JX)
      WL(1,26)=WL(1,50)
      YEDGE=YEDGE+DY
      DY=DY*2.0
      DESI=DESI*2.0
      DST=DST*2.0
      DEB=DEB*2.0
      WRITE(3,119) JMAX, DY
  119 FORMAT(/,3X,'NORMAL GRID POINTS REDUCED TO',I4,1X,/,
     *3X,'D(NORMAL) =',F14.6,/)
      GO TO 121
  151 WRITE(3,1030)
 1030 FORMAT(5X,'POINT ADDED AT EDGE',/)
      YEDGE=YEDGE+DY
      WL(1,JMAX)=WL(1,JMAX-1)
      UEL=WL(1,JMAX)
      IF(UEL.LE.0.) UEL=1.0
  121 DO 113 J=1,JMAX
  113 W(1,J)=WL(1,J)*UE/UEL
      GO TO 86
  115 FPE=-(2.*W(1,JMAX-1)-0.5*W(1,JMAX-2)-1.5*W(1,JMAX))/DY
      CFREXE=CFREX*FPE/FP
      IF(NT.EQ.1) GO TO 117
      IF((W(1,JMAX-1)/W(1,JMAX)).LT.0.9995) GO TO 150
      GO TO 118
  117 IF(CFREXE.GE.0.0005) GO TO 150
  118 CONTINUE
      DO 120 J=1,JMAX
      WL(1,J)=W(1,J)
  120 CONTINUE
      HL=H
      IF(CFREX.GT.0.0) GO TO 130
  137 SS=-CFI/DCFDS + SL
      IF(LR.EQ.0) GO TO 138
      XS=XL+DXDSL*(SS-SL)
      PS=(PL+DPDSL*(SS-SL))/DGR
      GO TO 124
  138 PSIS=SS/RREF/DGR
      WRITE(3,1081) PSIS
 1081 FORMAT(10X,'** FLOW SEPARATES FOR THIS STREAMLINE AT PSI =',F7.3
     A,1X,'DEGREES',/)
      GO TO 140
  124 WRITE(3,1080) XS,PS
 1080 FORMAT(10X,'** FLOW SEPARATES FOR THIS STREAMLINE AT X =',F7.4
     A,4X,'PHI =',F7.3,/)
      GO TO 140
  130 IF(LP.EQ.0) WRITE(3,1090) (W(1,J),J=1,JMAX)
```

```
      IF(IK.EQ.0) GO TO 131
      Y2D=Y(2)/DGR
      Y3D=Y(3)/DGR
      DPDS=DCPX*F(1)+DCPPH*F(2)
      WRITE(3,1050) Y(1),Y2D,Y3D,S,H,UE,DPDS,CFREX
      IF(KEC.EQ.1) WRITE(3,1052) CP,DCPX,DCPPH,DCPXE,D2PPH,R,ITER,
     A CFREXE,Y(4),Y(5)
  131 DCFDS=(CFREX-CFL)/(S-SL)
      CFI=CFREX
      SL=S
      XL=Y(1)
      PL=Y(2)
      DXDSL=F(1)
      DPDSL=F(2)
      IF(NHP.GE.MAXS) GO TO 140
      IF(Y(1).GT.BL) GO TO 140
      IF(IK.EQ.0) GO TO 40
      GO TO 550
  135 WRITE(3,136)
  136 FORMAT(/,3X,'LAST PROFILE FAILED TO CONVERGE AFTER',
     *1X,'40 ITERATIONS',/)
      GO TO 137
  142 WRITE(3,143)
  143 FORMAT(/,10X,'STREAMLINE TERMINATED - STEP SIZE REDUCED TO ',
     *'LOWER LIMIT',/)
  140 WRITE(3,141) NHP
  141 FORMAT(/,3X,'STREAMLINE TERMINATING AFTER',I4,1X,'STATIONS',/)
      KBS=KBS+1
      IF(KBS.LE.KBM) GO TO 8
      STOP
      END
C
C
C
C
```

16. Listing of Subroutine BGEOM

```
      SUBROUTINE BGEOM(XX,R,DRDX,GM,DGX)
C
C  THIS SUBROUTINE COMPUTES THE GEOMETRIC PROPERTIES
C  FOR A SPHERE-OGIVE-CYLINDER CONFIGURATION
C
C  X IS AXIAL POSITION
C  R IS BODY RADIUS
C  GEOMETRY FOR SPHERE-OGIVE-CYLINDER
      IF(XX.LE.0.1442) GO TO 10
      IF(XX.GE.5.19615) GO TO 20
      R=-13.+SQRT(14.**2-(XX-5.19615)**2)
      DRDX=(5.19615-XX)/(R+13.)
      D2RDX2=-(1.+DRDX**2)/(R+13.)
      GO TO 30
   10 R=SQRT(0.060811**2-(XX-0.1661438)**2)
      DRDX=(0.1661438-XX)/R
      D2RDX2=-(1.+DRDX**2)/R
      GO TO 30
   20 R=1.
      DRDX=0.
      D2RDX2=0.0
   30 GM=ATAN(DRDX)
```

```
      DGX=D2RDX2/(1.+DRDX**2)
      RETURN
      END
C
      SUBROUTINE BGEOM(X,R,DRDX,GM,DGX,TRAT)
C
C  GEOMETRY FOR AN ELLIPSOID OF REVOLUTION WITH THICKNESS
C  RATIO TRAT
C
      R=SQRT(TRAT**2*(1.0-(1.0-X)**2))
      DRDX=(1.0-X)*TRAT**2/R
      GM=ATAN(DRDX)
      D2RDX2=-TRAT**2/R**(1.0+(1.-X)*DRDX/R)
      DGX=D2RDX2/(1.0+DRDX**2)
      RETURN
      END
```

17. Listing of Subroutine BLOTNB

```
      SUBROUTINE BLCTNB(PSI,S,Y,F,DEXI,EXIH,BETA,DST)
      DIMENSION Y(6),F(6),T(4),P(4)
      COMMON /HALL/DSTN,DY,UE
      COMMON /OUTPT/R,CP,DCPX,DCPPH,DCPXP,D2PPH,GM,DGX
      COMMON /SCALE/HL,HLH,H
      COMMON /INTFC/XINT,PHIP,SAL,CAL,PM,PSIO,RPEB
      IF(PSI.LE.PSIO) KL=0
      GO TO (10,20,30), KL
      T(1)=0.0
      P(1)=0.0
      BQE=BPEB**2*RPEB
      EXIL=0.0
      KL=1
 10   T(4)=UE*H**2*RPEB
      P(4)=PSI
      DESI=PSI-P(1)
      DO 11 J=1,2
      JK=J+1
      P(JK)=P(1)+DPSI/4.0*2.0**(J-1)
      CALL SEHCAP(P(JK),CPB,UEB,DCP)
      IF(JK.NE.3) GO TO 11
      DCPS1=DCP
      UE3=UEE
 11   T(JK)=BQB*UEB*SIN(P(JK))**2
      Y(6)=EXIL+DPSI*(T(1)+4.*T(3)+T(4))/6.0
      EXIH=EXIL+DPSI*(T(1)+4.*T(2)+T(3))/12.0
      DEXI=Y(6)-EXIL
      BETA=-EXIH*DCPS1/(T(3)*UE3**2)
      EXIL=Y(6)
      P(1)=P(4)
      T(1)=T(4)
       UEL=UE
      IF(PSI.GE.PM) KL=2
      RETURN
30    XH=XL+DXL*DST/2.+DDXL*(DST/2.)**2/2.0
      XQ=XL+DXL*DST/4.+DDXL*(DST/4.)**2/2.0
      PH=PL+DPL*DST/2.+DDPL*(DST/2.)**2/2.0
      PQ=PL+DPL*DST/4.+DDPL*(DST/4.)**2/2.0
      CALL INVISD(XQ,PQ,UEQ,DPX,DPP,P,DPXP,DPP2)
      CALL INVISD(XH,PH,UEH,DPX,DPP,P,DPXP,DPP2)
      HH=(H+HL)/2.0
```

```
      HQ=HL+(H-HL)/4.0
      EXIH=EXIL+DST*(UEL*HL**2+4.*UEQ*HQ**2+UEH*HH**2)/12.0
      DXDS=DXL+DS1*DDXL/2.0
      DPDS=DPL+DST*DDPL/2.0
      BETA=-EXIH*(DPX*DXDS+DPP*DPDS)/((UEH*HH)**2*UEH)
      DEXI=Y(6)-EXIL
      EXIL=Y(6)
      UEL=UE
   20 XL=Y(1)
      PL=Y(2)
      DXL=F(1)
      DPL=F(2)
      DDXL=-SIN(Y(3))*COS(GM)*F(3)-COS(Y(3))*SIN(GM)*DGX*F(1)
      DDPL=COS(Y(3))*F(3)/R-DRDX*F(1)*SIN(Y(3))/R**2
      UEL=UE
      KL=3
      RETURN
      END
```

18. Listing of Subroutine FCN

```
      SUBROUTINE FCNJ(N,S,Y,PD)
      RETURN
      END
C
C
C
C
C
C
C
C
C
C
C
C
C
      SUBROUTINE FCN(N,S,Y,F)
      DIMENSION Y(N),F(N)
      COMMON /HALL/DSIN,DY,UE
      COMMON /OUTPT/R,CP,DCPX,DCPPH,DCPXP,D2PPH,GM,DGX
      COMMON /SCALE/HL,HLH,H
      CALL EGECM(Y(1),R,DRDX,GM,DGX)
      CALL INVISD(Y(1),Y(2),UE,DCPX,DCPPH,CP,DCPXP,D2PPH)
      CGM=COS(GM)
      SGM=SIN(GM)
      CTH=COS(Y(3))
      STH=SIN(Y(3))
      TTH=TAN(Y(3))
      F(1)=CTH*CGM
      F(2)=STH/R
      F(3)=0.5*(STH*CGM*DCPX+CTH*DCPPH/R)/UE**2-STH*SGM/R
      F(4)=-Y(4)*(Y(4)+SGM)/(R*CGM)
     ++0.5*(STH*CGM*DCPXP-CTH*D2PPH/R)/UE**2
     ++0.5*CGM*DCPX*Y(4)/(CTH*UE**2)
     ++0.5*DCPPH*(STH*CGM*DCPX-CTH*DCPPH/R)/(UE**2*UE**2)
      F(5)=Y(4)/(R*CTH)
      IF(N.EQ.5) RETURN
      H=S*CTH*EXP(Y(5))
      F(6)=UE*H**2
```

```
      RETURN
      END
C
C
C
C
C
C
C
C
C
C
C
C
C
C
```

19. Listing of Subroutine COEFF

```
      SUBROUTINE COEFF(JMAX,AM,BM,CM,DM,AS,CS,DS,W,WL,KB)
      COMMON /HALL/DSIN,DY,UE
      COMMON /BIG1/DETA,DEXI,EXIH,BIPH
      COMMON /SCALE/HL,HLH,H
      DIMENSION W(2,JMAX), WL(2,JMAX)
      DIMENSION AM(JMAX),BM(JMAX),CM(JMAX),DM(JMAX)
      DIMENSION AS(JMAX),DS(JMAX)
      IF(KB.EQ.2) GO TO 60
      IF(KB.EQ.3) GO TO 80
      BETA=0.5
C  SIMILAR SOLUTION COEFFICIENTS
      JM1=JMAX-1
      DO 20 J=2, JM1
      AM(J)=0.5*(1.0+0.5*BETA*W(2,J))
      BM(J)=1.0+(DETA**2)*BETA*W(1,J)
      CM(J)=1.0-AM(J)
      DM(J)=0.5*(DETA**2)*(BETA*(1.0+W(1,J)**2)+W(2,J)*0.5*
     *(W(1,J+1)-W(1,J-1))/DETA)
      AS(J)=0.25*DETA*(W(1,J+1)-W(1,J-1))
      DS(J)=0.0
  20  CONTINUE
      CS=0.5*DETA
      DS(JMAX)=0.0
      RETURN
C
  60  CONTINUE
      JM1=JMAX-1
      DO 65 J=2,JM1
      AM(J)=0.5+0.25*DETA*W(2,J)
      BM(J)=1.0+DETA**2*W(1,J)*(BIPH+2.*EXIH/DEXI)
      CM(J)=1.0-AM(J)
      DM(J)=0.5*(WL(1,J+1)-2.*WL(1,J)+WL(1,J-1))
     ++0.5*DETA**2*BIPH*(W(1,J)**2-WL(1,J)**2+2.)
     ++0.25*DETA*W(2,J)*(W(1,J+1)-W(1,J-1))
     ++DETA**2*EXIH/DEXI*(W(1,J)**2+WL(1,J)**2)
      AS(J)=0.25*DETA*(W(1,J+1)-W(1,J-1)+WL(1,J+1)-WL(1,J-1))
      DS(J)=DETA*(-0.25+EXIH/DEXI)*(WL(1,J)+WL(1,J-1))
  65  CONTINUE
      CS=DETA*(0.25+EXIH/DEXI)
      DS(JMAX)=DETA*(-0.25+EXIH/DEXI)*(WL(1,JMAX)+WL(1,JMAX-1))
      RETURN
C
```

```
C
C
C
C
C
C
C
C
C
C
C
C
C
C  COEFFICIENTS FOR HALL'S METHOD
  80  CONTINUE
      W(1,JMAX)=UE
      JM1=JMAX-1
      DO 90 J=2,JM1
      AM(J)=0.5/DY**2 + 0.25*W(2,J)/DY
      BM(J)=1.0/DY**2 + W(1,J)/DSTN
      CM(J)=0.5/DY**2 - W(2,J)*0.25/DY
      DM(J)=0.5/DY**2*(WL(1,J+1)-2.0*WL(1,J)+WL(1,J-1))
     ++(W(1,J)**2+WL(1,J)**2)*0.5/DSTN
     ++(W(1,JMAX)**2-WL(1,JMAX)**2)/(2.0*DSTN)
     ++0.25*W(2,J)/DY*(W(1,J+1)-W(1,J-1))
      ES(J)=DY*EL/(2.*DSTN*HLH)*(WL(1,J)+WL(1,J-1))
      AS(J)=0.25/DY*(W(1,J+1)-W(1,J-1)+WL(1,J+1)-WL(1,J-1))
  90  CONTINUE
      CS=DY*H/(2.*DSTN*HLH)
      DS(JMAX)=DY*HL/(2.*DSTN*HLH)*(WL(1,JMAX)+WL(1,JMAX-1))
      RETURN
      END
C
C
C
C
C
C
C
C
C
C
C
C
C
C
C
```

20. Listing of Subroutine INVERT

```
      SUBROUTINE INVERT(JMAX,A,B,C,D,AS,CS,DS,W,KB)
      DIMENSION A(JMAX), B(JMAX),C(JMAX),D(JMAX)
      DIMENSION AS(JMAX), DS(JMAX)
      DIMENSION W(2,JMAX)
      DIMENSION E(50), EL(50), G(50)
      COMMON /HALL/DX,DY,UE
      JM2=JMAX-2
      E(JMAX)=0.0
      G(JMAX)=0.0
      EL(JMAX)=UE
      IF(KB.NE.3)  EL(JMAX)=1.0
```

```
      DO 20 JK=1, JM2
      J=JMAX-JK
      DEN=B(J)-C(J)*E(J+1)+CS*(C(J)*G(J+1)-AS(J))
      E(J) = (A(J)-CS*(C(J)*G(J+1)-AS(J)))/DEN
      G(J) = (C(J)*G(J+1)-AS(J))/DEN
      EL(J) = (D(J)+(C(J)*G(J+1)-AS(J))*DS(J)+C(J)*EL(J+1))/DEN
   20 CONTINUE
      DO 10 J=2,JMAX
      W(1,J)=E(J)*W(1,J-1)+G(J)*W(2,J-1)+EL(J)
      W(2,J)=W(2,J-1)-CS*(W(1,J)+W(1,J-1))+DS(J)
   10 CONTINUE
      RETURN
      END
C
C
C
C
C
C
C
C
C
C
```

21. Listing of Function KRUNGE

```
      FUNCTION KRUNGE(Y,F,X,H,N,HMAX,MF)
      DIMENSION PHI(12),SAVEY(12),Y1(12),Y2(12),YKP(12),FKP(12),Y(N)
     *F(N)
      DATA M,LOCF,EPS/0,0,5.E-4/
      M=M+1
      GO TO (5,45,65,85),M
    5 IF (LOCF.GT.0) GO TO 25
      IF (MF.EQ.1) GO TO 205
      IF (ABS(H).GE.ABS(HMAX)) H=HMAX
      DO 15 J=1,N
      YKP(J)=Y(J)
   15 FKP(J)=F(J)
      XO=X
   25 DO 35 J=1,N
      SAVEY(J)=Y(J)
      PHI(J)=F(J)
   35 Y(J)=SAVEY(J)+0.5*H*F(J)
      X=X+0.5*H
      KRUNGE=1
      RETURN
   45 DO 55 J=1,N
      PHI(J)=PHI(J)+2.0*F(J)
   55 Y(J)=SAVEY(J)+0.5*H*F(J)
      RETURN
   65 DO 75 J=1,N
      PHI(J)=PHI(J)+2.0*F(J)
   75 Y(J)=SAVEY(J)+H*F(J)
      X=X+0.5*H
      RETURN
   85 DO 95 J=1,N
   95 Y(J)=SAVEY(J)+(PHI(J)+F(J))*H/6.0
      IF (MF.EQ.1) GO TO 165
      IF (LOCF-1)  105,125,145
  105 DO 115 J=1,N
      Y2(J)=Y(J)
```

```
      F(J)=FKE(J)
  115 Y(J)=YKE(J)
      X=XO
      H=H/2.
      M=1
      LOCP=1
      GO TO 25
  125 DO 135 J=1,N
  135 Y1(J)=Y(J)
      LOCP=2
      M=0
      RETURN
  145 DO 155 J=1,N
      IF (ABS(Y(J)).LT.5.E-04) GO TO 155
      ER=(Y(J)-Y2(J))/Y(J)
      IF (ABS(ER)-EPS)  155,155,175
  155 CONTINUE
  165 H=2.*H
      MR=0
      LOCP=0
      KRUNGE=0
      M=0
      RETURN
  175 DO 185 J=1,N
      Y(J)=YKP(J)
      F(J)=FKP(J)
  185 Y2(J)=Y1(J)
      X=XO
      H=H/2.
      IF (ABS(H).LT.1.E-10) GO TO 195
      LOCP=1
      M=1
      GO TO 25
  195 KRUNGE=2
      M=0
      LOCP=0
      RETURN
  205 DO 215 J=1,N
      Y(J)=YKP(J)
  215 F(J)=FKP(J)
      X=XC
      H=H/2.
      GO TO 25
      END
C
C
C
C
C
```

22. Listing of Subroutine MILNES

```
      SUBROUTINE MILNES(Y,F,PCM,NDE,DS,S)
      DIMENSION Y(6),F(6),PCM(6,4,2)
    5 DO 10 J=1, NDE
      Y(J)=PCM(J,4,1)+4.*DS/3.*(2.*PCM(J,1,2)-PCM(J,2,2)
     ++2.*PCM(J,3,2))
   10 CONTINUE
      CALL FCN(NDE,S,Y,F)
      DO 30 J=1, NDE
```

```
      Y(J)=PCM(J,2,1)+DS/3.*(P(J)+4.*PCM(J,1,2)+PCM(J,2,2))
   30 CONTINUE
      CALL PCM(NDE,S,Y,P)
      DO 50 J=1, NDE
      DO 60 JJ=1,3
      JK=5-JJ
      JK2=JK-1
      DO 70 JL=1,2
   70 PCM(J,JK,JL)=PCM(J,JK2,JL)
   60 CONTINUE
      PCM(J,1,1)=Y(J)
   50 PCM(J,1,2)=P(J)
      S=S+DS
      RETURN
      END
C
C
C
```

23. Listing of Subroutine PRESS

```
      SUBROUTINE PRESS
      COMMON /PDATA/PHI(40),X(40),CP(40,40),CPX(2,40)
      COMMON /CESTA/NAS,NCS
C
C
C  NAS = NUMBER OF AXIAL PRESSURE STATIONS
C  NCS = NUMBER OF CIRCUMFERENTIAL PRESSURE STATIONS
C
      DGR=ACOS(-1.)/180.
C
      READ(1,5) NCS,NAS
    5 FORMAT(I2,1X,I2)
      DO 10 J=1,NCS
      READ(1,15) PHI(J)
      PHI(J)=PHI(J)*DGR
   15 FORMAT(F10.5)
   10 CONTINUE
C
      DO 20 J=1,NAS
      READ(1,15) X(J)
   20 CONTINUE
C
      DO 30 J=1,NAS
      READ(1,35) (CP(J,K), K=1,NCS)
   35 FORMAT(5(1X,F10.5))
   30 CONTINUE
C
C
C  CPX(1,J) AND CPX(2,J) ARE THE AXIAL DERIVATIVES OF THE PRESSURE
C  COEFFICIENT AT THE INTERFACE AND BODY END.  EACH CIRCUMFERENTIAL
C  PLANE MUST HAVE A SET AS INPUT TO THE QUADRATIC SPLINE ROUTINE.
C
      DO 40 J=1,NCS
      READ(1,16) CPX(1,J),CPX(2,J)
   16 FORMAT(F10.5,1X,F10.5)
   40 CONTINUE
      RETURN
      END
C
```

24. Listing of Subroutine SPHCAP

```
C
C
      SUBROUTINE SPHCAP(P,CP,UE,DCPSI)
      DIMENSION COE(10)
      DATA CCE/0.666823, 1.130572, -0.134360, 0.131321, -0.009964,
     * 0.014346, 0.018485, -0.003913, 0.014531, -0.827853/
      COMMON /INTFC/XINT,PHIP,SAL,CAL,PM,PSIO,RPER
      IF(P.EQ.0.0) M=0
      GO TO (210,200),M
      CALL INVISD(XINT,PHIP,UE,DCPX2,DCPPH2,CP2,DCPXP,D2PPH)
      CALL EGECM(XINT,R,DR,GM,DGX)
      DPSIDX=(SAL*SIN(GM)*COS(PHIP)-CAL*COS(GM))*DGX/SIN(PM)
      DCPDSI=0.0
      CP=0.0
      DO 10 J=1,9
      CP=CP+CCE(J)*COS(J*PM)
   10 DCPDSI=DCPDSI-J*COE(J)*SIN(J*PM)
      CP=CP+CCE(10)
      DCPX=DCPDSI*DPSIDX
      WRITE(3,20) CP,CP2,DCPX,DCPX2
   20 FORMAT(/,4X,'CONTINUITY OF PRESSURE AND DERIVATIVE ACROSS',
     A' INTERFACE: ',/,2(5X,F12.5),/,2(5X,F12.5),/)
      M=2
      IF(ALS(CP-CP2).GT.0.003.OR.ABS(DCPX-DCPX2).GT.4.0) M=1
      IF(M.EQ.1) WRITE(3,30)
   30 FORMAT(4X,'***** QUADRATIC SPLINE EXTENDED TO INCLUDE',
     A' NOSE REGION *****',/)
      IF(M.EQ.2) RETURN
      PM=PM/2.0
      CPP=DCPX2/DPSIDX
      Y=CP2/2.0+0.5-CPP*PM/8.0
      YP1=8.0*(Y-1.0)/PM**2
      YP2=8.0*(Y-CP2+CPP*PM/2.0)/PM**2
      RETURN
  210 IF(P.GT.PM) GO TO 220
      CP=1.0+YP1*P**2/2.0
      DCPSI=YP1*P
      GO TO 230
  220 CP=CP2+CPP*(P-PM)+YP2*(P-PM)**2/2.0
      DCPSI=CPP+YP2*(P-PM)
  230 UE=SQRT(1.0-CP)
      RETURN
  200 CP=0.0
      DCPSI=0.0
      DO 250 J=1,9
      DCPSI=DCPSI-J*COE(J)*SIN(J*P)
  250 CP=CP+CCE(J)*COS(J*P)
      CP=CP+CCE(10)
      UE=SQRT(1.0-CP)
      RETURN
      END
C
C
C
C
C
C
```

```
C
C
C
C
```

25. Listing of Subroutine MIDPTS

```
      SUBROUTINE MIDPTS(AA,BB,CC,D,YM,N)
      DIMENSION G(40),U(40),AA(40),BB(40),CC(40),D(40),YM(40)
      NM1=N-1
      NM2=N-2
      G(1)=-CC(1)/BB(1)
      U(1)=D(1)/BB(1)
      DO 5 K=2,NM1
      ALPHA=AA(K)*G(K-1)+BB(K)
      G(K)=-CC(K)/ALPHA
    5 U(K)=(D(K)-AA(K)*U(K-1))/ALPHA
      YM(NM1)=U(NM1)
      DO 10 K=1,NM2
      JK=NM2+1-K
   10 YM(JK)=G(JK)*YM(JK+1)+U(JK)
      RETURN
      END
```

```
C
C
C
C
C
C
C
C
C
C
C
C
C
C
C
C
```

26. Listing of Subroutine INVISD

```
      SUBROUTINE INVISD(XX,PPH,UE,DCPX,DCPPH,CPC,DCPXP,D2PPH)
      COMMON /PDATA/PHI(40),X(40),CP(40,40),CPX(2,40)
      COMMON /CESTA/NAS,NCS
      DIMENSION DELX(40),DELP(40),DP(40),CP1(40,40),CP2(40,40)
      DIMENSION A(40),B(40),C(40),D(40),YM(40),CPT(2,40)
      DIMENSION ALP(40),BETA(40),GAM(40),DEL(40)
      DATA M/0/
      IF(M.NE.0) GO TO 100
C
C
C
      DO 10 J=2,NAS
   10 DELX(J)=X(J)-X(J-1)
      NCM1=NCS-1
      NCM2=NCS-2
      NAM1=NAS-1
      NAM2=NAS-2
      A(1)=0.0
```

```
      B(1)=2./DELX(2)+(2.+DELX(3)/DELX(2))/(DELX(3)+DELX(2))
      C(1)=DELX(2)/DELX(3)/(DELX(2)+DELX(3))
      DO 20 N=2,NAM2
      A(N)=DELX(N+1)/DELX(N)/(DELX(N+1)+DELX(N))
      B(N)=(2.+DELX(N)/DELX(N+1))/(DELX(N)+DELX(N+1))
     ++(2.+DELX(N+2)/DELX(N+1))/(DELX(N+2)+DELX(N+1))
      C(N)=DELX(N+1)/DELX(N+2)/(DELX(N+1)+DELX(N+2))
   20 CONTINUE
      A(NAM1)=DELX(NAS)/DELX(NAM1)/(DELX(NAS)+DELX(NAM1))
      B(NAM1)=(2.+DELX(NAM1)/DELX(NAS))/(DELX(NAS)+DELX(NAM1))
     ++2./DELX(NAS)
      C(NAM1)=0.0
C
C
      DO 30 N=1,NCS
      D(1)=CP(1,N)*2./DELX(2)+CP(2,N)*((2.+DELX(3)/DELX(2))/(DELX(3)
     ++DELX(2))+DELX(2)/DELX(3)/(DELX(2)+DELX(3)))  + CPX(1,N)/2.0
      DO 32 J=2,NAM2
      D(J)=CP(J,N)*(A(J)+(2.+DELX(J)/DELX(J+1))/(DELX(J)+DELX(J+1)))
     ++CP(J+1,N)*(C(J)+(2.+DELX(J+2)/DELX(J+1))/(DELX(J+2)+DELX(J+1)))
   32 CONTINUE
      D(NAM1)=CP(NAM1,N)*(DELX(NAS)/DELX(NAM1)/(DELX(NAS)+DELX(NAM1))
     ++(2.+DELX(NAM1)/DELX(NAS))/(DELX(NAM1)+DELX(NAS)))
     ++CP(NAS,N)*2./DELX(NAS)  - CPX(2,N)/2.0
      CALL MIDPTS(A,B,C,D,YM,NAS)
      DO 34 J=2,NAM1
      T1=DELX(J+1)+DELX(J)
      T2=(YM(J)-CP(J,N))/DELX(J+1)
      T3=(YM(J-1)-CP(J,N))/DELX(J)
      CP1(J,N)=2./T1*(DELX(J)*T2-DELX(J+1)*T3)
      CP2(J,N)=8./T1*(T2+T3)
   34 CONTINUE
      CP1(1,N)=CPX(1,N)
      CP2(1,N)=8.*(YM(1)-CP(1,N)-CPX(1,N)*DELX(2)/2.)/DELX(2)**2
      CP1(NAS,N)=CPX(2,N)
      CP2(NAS,N)=8.*(YM(NAM1)-CP(NAS,N)+CPX(2,N)*DELX(NAS)/2.)/
     ADELX(NAS)**2
   30 CONTINUE
C
C
      DO 35 J=1,NAM1
   35 DELX(J)=(X(J+1)+X(J))/2.
C
      DO 40 J=2,NCS
   40 DELP(J)=PHI(J)-PHI(J-1)
C
      ALP(1)=0.0
      BETA(1)=2./DELP(2)
      GAM(1)=(2.+DELP(3)/DELP(2))/(DELP(3)+DELP(2))
      DEL(1)=DELP(2)/DELP(3)/(DELP(2)+DELP(3))
      A(1)=ALP(1)
      B(1)=BETA(1)+GAM(1)
      C(1)=DEL(1)
      DO 45 J=2,NCM2
      ALP(J)=DELP(J+1)/DELP(J)/(DELP(J+1)+DELP(J))
      BETA(J)=(2.+DELP(J)/DELP(J+1))/(DELP(J)+DELP(J+1))
      GAM(J)=(2.+DELP(J+2)/DELP(J+1))/(DELP(J+2)+DELP(J+1))
      DEL(J)=DELP(J+1)/DELP(J+2)/(DELP(J+1)+DELP(J+2))
      A(J)=ALP(J)
      B(J)=BETA(J)+GAM(J)
      C(J)=DEL(J)
```

```
   45  CONTINUE
       ALP(NCM1)=DELP(NCS)/DELP(NCM1)/(DELP(NCS)+DELP(NCM1))
       BETA(NCM1)=(2.+DELP(NCM1)/DELP(NCS))/(DELP(NCM1)+DELP(NCS))
       GAM(NCM1)=2./DELP(NCS)
       DEL(NCM1)=0.0
       A(NCM1)=ALP(NCM1)
       B(NCM1)=BETA(NCM1)+GAM(NCM1)
       C(NCM1)=DEL(NCM1)
C
C
       DO 50 J=1,NCM1
   50  DP(J)=(PHI(J+1)+PHI(J))/2.0
       M=1
C
C
C
C
C
  100  DO 105 J=1,NAM1
       IF(XX.LE.DELX(J)) GO TO 110
  105  CONTINUE
  110  IF(XX.GT.DELX(NAM1)) J=NAS
       DEX=XX-X(J)
       DO 115 K=1,NCS
       CPT(1,K)=CP(J,K)+CP1(J,K)*DEX+CP2(J,K)*DEX**2/2.
  115  CPT(2,K)=CP1(J,K)+CP2(J,K)*DEX
C
C  SPLINE FIT BOTH CP AND D(CP)/D(X)
C
       DO 130 L=1,NCM1
       IF(PPH.LE.DP(L)) GO TO 140
  130  CONTINUE
  140  IF(PPH.GT.DP(NCM1)) L=NCS
       DPHI=PPH-PHI(L)
C
       DO 120 N=1,2
       D(1)=CPT(N,1)*BETA(1)+CPT(N,2)*(GAM(1)+DEL(1))
       DO 125 K=2,NCM2
  125  D(K)=CPT(N,K)*(ALP(K)+BETA(K))+CPT(N,K+1)*(GAM(K)+DEL(K))
       D(NCM1)=CPT(N,NCM1)*(ALP(NCM1)+BETA(NCM1))+CPT(N,NCS)*GAM(NCM1)
       CALL MIDPTS(A,B,C,D,YM,NCS)
       IF(L.EQ.1) GO TO 126
       IF(L.EQ.NCS) GO TO 127
       T1=DELP(L+1)+DELP(L)
       T2=(YM(L)-CPT(N,L))/DELP(L+1)
       T3=(YM(L-1)-CPT(N,L))/DELP(L)
       TJ=2./T1*(DELP(L)*T2-DELP(L+1)*T3)
       TK=8./T1*(T2+T3)
       GO TO 128
  126  TJ=0.0
       TK=8.*(YM(1)-CPT(N,L))/DELP(2)**2
       GO TO 128
  127  TJ=0.0
       TK=8.*(YM(NCM1)-CPT(N,L))/DELP(NCS)**2
  128  CPT(N,1)=CPT(N,L)+TJ*DPHI+TK*DPHI**2/2.
       GO TO (122,124),N
  122  IF(PPE.GT.0.) GO TO 123
       DCPPH=0.0
       D2PPH=TK
       GO TO 120
  123  DCPPH=TJ+TK*DPHI
```

```
      D2PPB=TK
      GO TO 120
  124 DCPXP=TJ+TK*DPHI
  120 CONTINUE
      CPC=CPT(1,1)
      DCPX=CPT(2,1)
      UE=SQRT(1.-CPC)
      RETURN
      END
C
C
C
      SUBROUTINE INVISD(X,PHI,CP,UE,UEX,UEP)
C
C  POTENTIAL SOLUTION FOR ELLIPSOID OF REVOLUTION
C
      COMMON /STAG/XO,YO,SAL,CAL,TRAT,SC,CBE,SBE,CGO,AO,BO,VGRAD
      DATA M/0/
      IF(M.EQ.1) GO TO 10
      T1=2.0*CAL/(2.0-AO)
      T2=2.0*SAL/(2.0-BO)
  10  CALL RGECM(X,R,DRDX,GM,DGX,TRAT)
      APSQ=(X-1.0)**2+(R/TRAT**2)**2
      T5=(T1*R/TRAT**2+T2*(X-1.0)*COS(PHI))
      UE=SQRT(T5**2/APSQ+(T2*SIN(PHI))**2)
      CP=1.0-UE**2
      UEX=(T5*(T1*DRDX/TRAT**2+T2*COS(PHI))/APSQ-T5**2*((X-1.0)+
     A R*DRDX/TRAT**2/TRAT**2)/APSQ**2)/UE
      UEP=(-T5*(T2*(X-1.0)*SIN(PHI))/APSQ+T2**2*SIN(PHI)*COS(PHI))/UE
      IF(M.EQ.1) RETURN
      DUSDX=(T1*DRDX/TRAT**2+T2)/SQRT(APSQ)
      VGRAD=DUSDX*CGO/T2*YO
      M=1
      RETURN
      END
```

27. Listing of Subroutine STAGN

```
      SUBROUTINE STAGN(ALP,XO,XNOSE,RPER)
C  THIS SUBROUTINE COMPUTES THE NEWTONIAN STAGNATION POINT FOR
C  A GIVEN EFFECTIVE ANGLE OF ATTACK
      XC=XNOSE+RPER*(1.0-COS(ALP))
      RETURN
      END
      SUBROUTINE STAGN(ALP,XO,YO,TRAT,AC,BO)
C
C  THIS ROUTINE LOCATES THE STAGNATION POINT ON
C  AN ELLIPSOID OF REVOLUTION
C
      E=SQRT(1.0-TRAT**2)
      ESQ=E*E
      EQB=ESQ*E
      AC=2.0*(1.0-ESQ)*(0.5*ALOG((1.+E)/(1.-E))-E)/EQB
      BC=1.0/ESQ-(1.0-ESQ)*ALOG((1.+E)/(1.-E))/(2.0*EQB)
      P=(2.0-AO)*TAN(ALP)/(2.0-BO)
      YO=P*TRAT**2/SQRT(1.0+(P*TRAT)**2)
      XC=1.0-SQRT(1.0-(YO/TRAT)**2)
      RETURN
      END
```

28. Listing of Input Data for Sphere-Cylive-Cylinder Configuration

```
 0.060811  0.1053328     17.50          0.1442
 0.0010           2.0            65.0
 000 000 001 003 300 000 000 000 002 000 000
10 34
0.0
11.25
33.75
52.25
78.75
101.25
123.75
146.25
168.75
180.00
0.1442
0.15690
0.16
0.17
0.18
0.20
0.21630
0.26620
0.32870
0.40964
0.50
0.60
0.70
0.80
0.90
1.00
1.100
1.20
1.50
1.75
2.0
2.5
3.0
3.5
4.0
4.5
4.61
5.66
6.97
8.29
9.61
12.24
14.87
17.5
 0.98200000 0.85800000 0.16400000 -0.4890000 -1.3840000
 -1.658      -1.588      -1.850      -1.638      -1.598
 0.9718      0.8890      0.2320      -0.7503     -1.5789
 -1.8947     -1.6459     -1.1143     -0.6985     -0.6465
 0.9703      0.8933      0.2408      -0.7800     -1.610
 -1.9010     -1.6360     -1.09       -0.695      -0.639
 0.9665      0.8989      0.2579      -0.8450     -1.6620
 -1.910      -1.604      -1.012      -0.672      -0.614
 0.965       0.900       0.261       -0.8540     -1.664
```

-1.917	-1.580	-0.965	-0.653	-0.595
0.9701	C.9030	0.2403	-0.8330	-1.6310
-1.891	-1.5440	-0.904	-0.619	-0.560
0.9770	C.9047	0.2193	-0.8100	-1.6163
-1.8723	-1.5214	-0.8717	-0.5950	-0.5380
0.9890	C.9040	0.2242	-0.7777	-1.5840
-1.8328	-1.4710	-0.8134	-0.5300	-0.4850
0.9862	C.9027	0.2347	-0.7498	-1.5430
-1.7862	-1.4304	-0.784	-0.4700	-0.4300
0.9811	C.8995	0.2460	-0.7169	-1.4910
-1.7303	-1.3819	-0.7492	-0.4200	-0.3820
0.975	C.894	0.256	-0.68	-1.432
-1.7	-1.329	-0.725	-0.382	-0.345
0.967	C.888	0.262	-0.656	-1.398
-1.666	-1.282	-0.694	-0.354	-0.317
0.958	C.882	0.267	-0.639	-1.374
-1.636	-1.235	-0.667	-0.335	-0.300
0.95	C.874	0.269	-0.626	-1.355
-1.608	-1.190	-0.636	-0.321	-0.284
0.942	C.867	0.271	-0.615	-1.34
-1.583	-1.147	-0.608	-0.314	-0.277
0.935	C.859	0.272	-0.606	-1.326
-1.559	-1.105	-0.580	-0.309	-0.274
0.926	C.851	0.271	-0.597	-1.315
-1.536	-1.068	-0.552	-0.305	-0.272
0.918	C.842	0.270	-0.592	-1.306
-1.516	-1.035	-0.525	-0.300	-0.269
0.894	C.815	0.259	-0.580	-1.287
-1.467	-0.950	-0.450	-0.286	-0.264
0.872	0.794	0.243	-0.575	-1.278
-1.435	-0.893	-0.391	-0.274	-0.258
0.849	C.770	0.223	-0.572	-1.272
-1.412	-0.834	-0.337	-0.262	-0.254
0.798	C.720	0.178	-0.574	-1.264
-1.384	-0.721	-0.253	-0.245	-0.245
0.741	C.666	0.129	-0.582	-1.264
-1.347	-0.624	-0.194	-0.205	-0.224
0.684	C.613	0.080	-0.595	-1.277
-1.284	-0.545	-0.154	-0.178	-0.177
0.627	C.563	0.033	-0.610	-1.290
-1.199	-0.477	-0.162	-0.156	-0.148
0.583	C.518	-0.011	-0.627	-1.299
-1.105	-0.415	-0.215	-0.147	-0.137
0.5570	C.5100	-0.0196	-0.6305	-1.3070
-1.083	-0.3964	-0.23	-0.150	-0.140
0.5441	C.4730	-0.0528	-0.6705	-1.3163
-0.860	-0.39	-0.425	-0.4477	-0.450
0.5380	C.4639	-0.0790	-0.7184	-1.2856
-0.7187	-0.4822	-0.7553	-0.8458	-0.85
0.531	0.4526	-0.1006	-0.7546	-1.1897
-0.609	-0.5472	-0.8586	-1.0870	-1.117
0.525	C.4465	-0.1148	-0.7658	-1.2440
-0.84	-0.635	-0.858	-1.1896	-1.240
0.5144	C.4328	-0.1205	-0.7538	-1.4508
-1.067	-0.6463	-0.6351	-0.7362	-0.750
0.5080	C.4326	-0.0746	-0.6380	-1.0941
-0.54	-0.32	-0.38	-0.4936	-0.509
0.5012	C.4354	-0.0181	-0.5321	-0.8233
-0.415	-0.280	-0.340	-0.3949	-0.402
-1.480	-C.01225			
4.60	C.00C74			

3.40	C.02370
-16.67	C.0480
-15.0	C.0804
-2.20	C.01683
15.0	-C.01633
32.25	-C.00483
47.5	C.0130
53.25	C.01833
50.0	
75.0	

U S GOVERNMENT PRINTING OFFICE 1969 - 648-034/00380